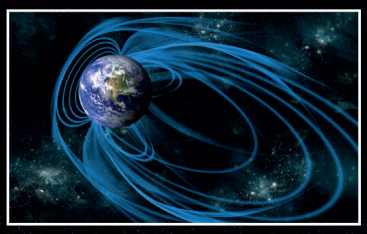
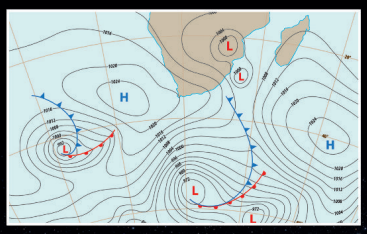
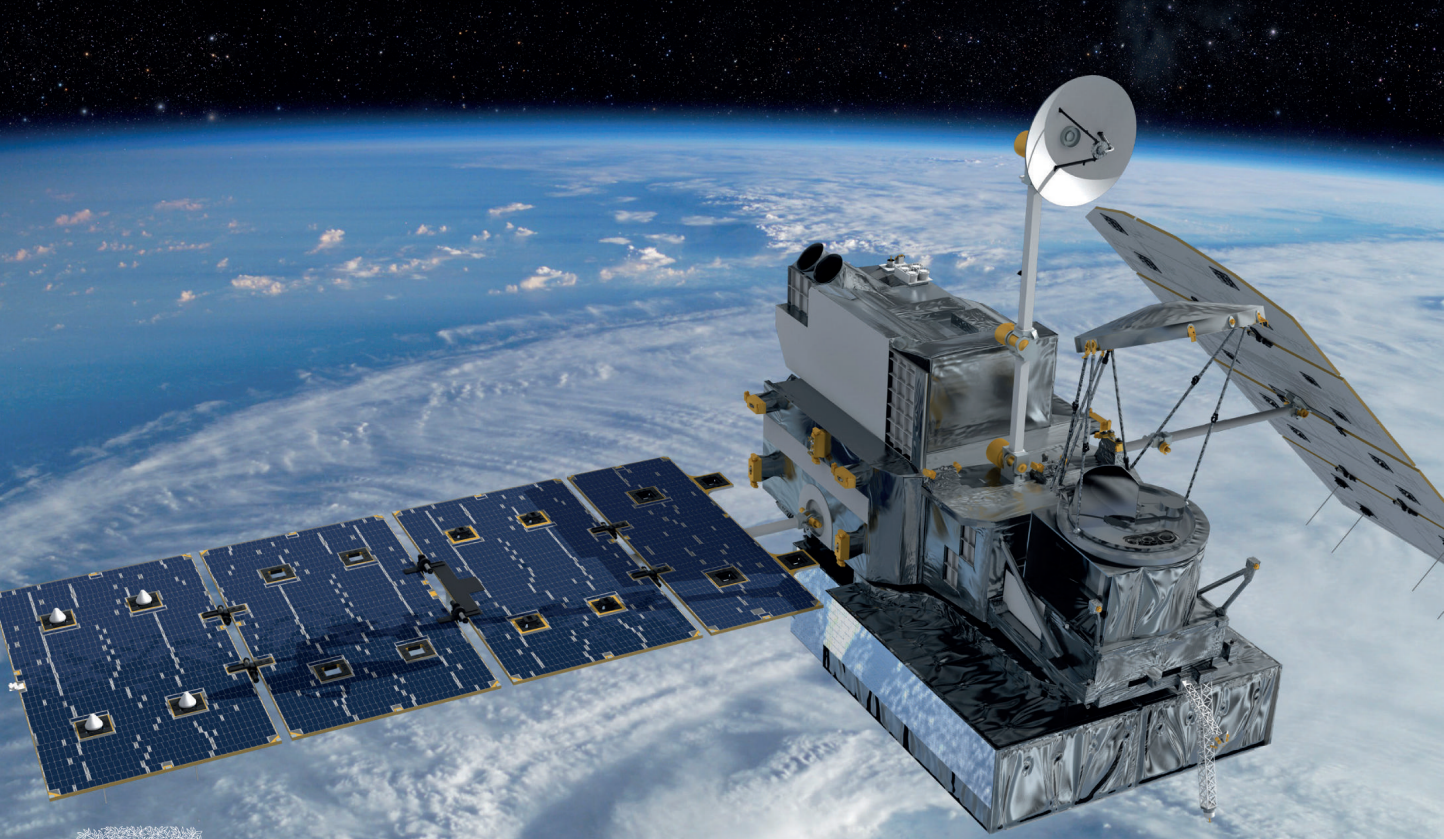
SECOND EDITION

DATA ASSIMILATION FOR THE GEOSCIENCES

FROM THEORY TO APPLICATION







STEVEN J. FLETCHER

ELSEVIER

Data Assimilation for the Geosciences

From Theory to Application

This page intentionally left blank

Data Assimilation for the Geosciences

From Theory to Application

Second Edition

Steven J. Fletcher
Cooperative Institute for Research in the Atmosphere (CIRA)
Colorado State University
Fort Collins, CO, United States

Elsevier
Radarweg 29, PO Box 211, 1000 AE Amsterdam, Netherlands
The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, United Kingdom
50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States

ISBN: 978-0-323-91720-9

For information on all Elsevier publications
visit our website at https://www.elsevier.com/books-and-journals

Publisher: Candice Janco
Acquisitions Editor: Amy Shapiro
Editorial Project Manager: Tim Eslava
Production Project Manager: Paul Prasad Chandramohan
Cover Designer: Mark Rogers

Typeset by VTeX

Contents

CHAPTER

Introduction 1

Data assimilation plays a vital role in how forecasts are made for different geophysical disciplines. While the numerical model of the geophysical systems are critical, so are the observations of the same system, be they direct or indirect. Neither the model nor the observations are perfect, and both are required for an improved forecast than can be achieved through solely using the numerical model without guidance of how accurate the current state is, or through producing a persistence, or advection, forecast from observations.

Fig. 1.1 shows a conceptual diagram of the forecast skill of different methods without data assimilation from [300]. We see that the order in which the forecast skill drops off, at least for the atmosphere and the prediction of clouds, is quite telling. Data assimilation, when included in this type of figure, will have a higher forecast skill, and for longer than the other approaches. Why? Because when I am asked what I do at Colorado State University, I say that I do research in data assimilation, to which the usual response is "What's that?" I reply: "It is the science of combining the numerical models and observations of a geophysical system, such that the new forecast produced by the numerical model is better than the model or the observations on their own."

Since the first edition of this textbook there has been a very detailed commissioned manuscript for the Journal of Advances in Modeling Earth Systems (JAMES) entitled *Confronting the Challenge of Modeling Cloud and Precipitation Microphysics*, [306] where there are two figures that we have copies of in Fig. 1.2 and Fig. 1.3 that show a schematic of the different process and scales associated with cloud prediction, and a schematic of the scales associated with atmospheric and climate prediction, highlighting the challenge that data assimilation has to face.

Data assimilation acts as a bridge between numerical models and observations. There are many different methods to enable the bridging, but while data assimilation was used initially in engineering, it should not be considered as just an engineering tool. Its application today in numerical weather prediction, numerical ocean prediction, land surface processes, hydrological cycle prediction, cryosphere prediction, space weather prediction, soil moisture evolution, land surface-atmosphere coupling, ocean-atmosphere coupling, carbon cycle prediction, and climate reanalysis, to name but a few areas of application, require many different techniques and approaches to deal with the dimension of the problems, the different time scales for different geophysical processes, nonlinearity of the observations operators, and non-Gaussianity of the distribution for the errors associated with the different processes and observation types.

The weather forecasts that you see on television, read on a phone, tablet or computer, or hear on the radio are generated from the output of a data assimilation system. Water resource managers rely on forecasts from a cryospheric-driven data assimilation system. Data assimilation plays an important part in renewable energy production. For example, wind farms require advance knowledge of *ramp-up and ramp-down* events. These are times when the wind is forecasted to exceed the safe upper limit for which

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00005-X

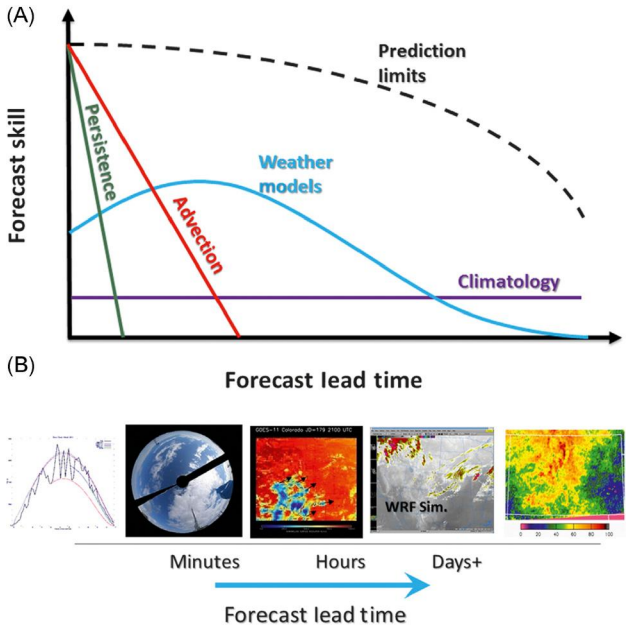


FIGURE 1.1

Copy of Fig. 1 from [300] of the different forecast skill lengths for persistence, advection, numerical modeling, and climatology.

the turbines can operate without overloading the blade motors. There were cases in the United Kingdom where ramp-up events were not correctly forecasted; the wind turbines were overloaded, caught fire and exploded as a result. Wind farms need to inform the electrical grids how much electricity they can provide either that day or over the next 48 hours. If the wind is forecasted to exceed the safe speed, then wind farms have to know how long the turbines will be switched off for, and how much less electricity they can provide.

Solar farms have a similar issue with clouds and snow/ice/dust. While individual clouds are difficult to predict, a general idea of cloud percentage cover is important for the amount of electricity that a solar farm can provide. Snow forecasts are important for solar farms that are in areas of plentiful sunlight but receive snow in the winter months. The forecast of Haboobs in the desert regions is also important for advance knowledge of the possibility of being able to produce electricity from solar panels. Knowing whether panels are covered in dust, or advance warning that they may be covered in dust and sand, is important. The forecasts are a by-product of a data assimilation system.

Data assimilation is also known in some scientific disciplines as inverse modeling. This is where we may not be producing a forecast, but we wish to combine an a priori state with an observation that is not directly of the a priori variables, to extract an estimate of the physical state at that time and place. A very frequent use of this technique is referred to as **retrieval**. In some geophysical fields, and for certain applications, the *retrieved product* may be assimilated into the model, rather than assimilating the indirect observation itself. This practice was quite common in the early stages of satellite data

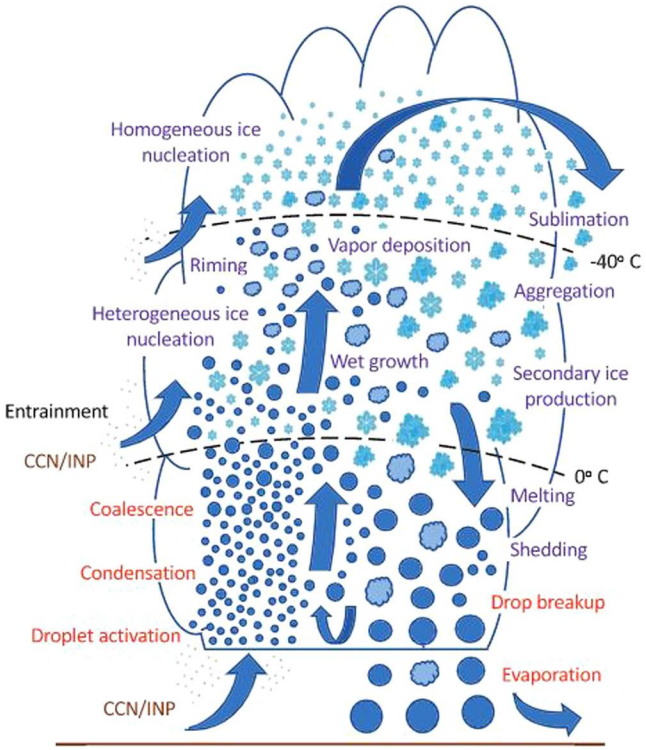


FIGURE 1.2

Schematic of the processes in cloud formation and stainability; Fig. 1 from Morrison, H., van Lier-Walqui, M., Fridlind, A. M., Grabowski, W. W., Harrington, J. Y., Hoose, C., et al. (2020). Confronting the challenge of modeling cloud and precipitation microphysics. Journal of Advances in Modeling Earth Systems, 12, e2019MS001689. https://doi.org/10.1029/2019MS001689, https://creativecommons.org/licenses/.

assimilation, as it meant that it avoided the highly nonlinear Jacobians of what are called **radiative transfer models** in the minimization of the **cost function**. Retrievals also play a vital part in gaining information from satellite **brightness temperatures** and **radiances**.

There are many different forms of data assimilation spanning many decades and uses, and each system has its advantages and disadvantages. The earliest forms of data assimilation were referred to as **objective analysis** and included empirical methods, where there is no probabilistic information determining the weights given to observations. These early data assimilation approaches were also referred to as **successive correction methods**, where they applied a series of corrections at different scales to filter the unresolved scales. Examples of these successive correction schemes are the Cressman and the Barnes schemes [24,79].

The next set of data assimilation methods after the successive correction methods were the different versions of **optimum interpolation**. The basis of OI, as it is more commonly known nowadays, is the minimization of a least squares problem. The first appearance of OI was in Gandin's 1963 book (in

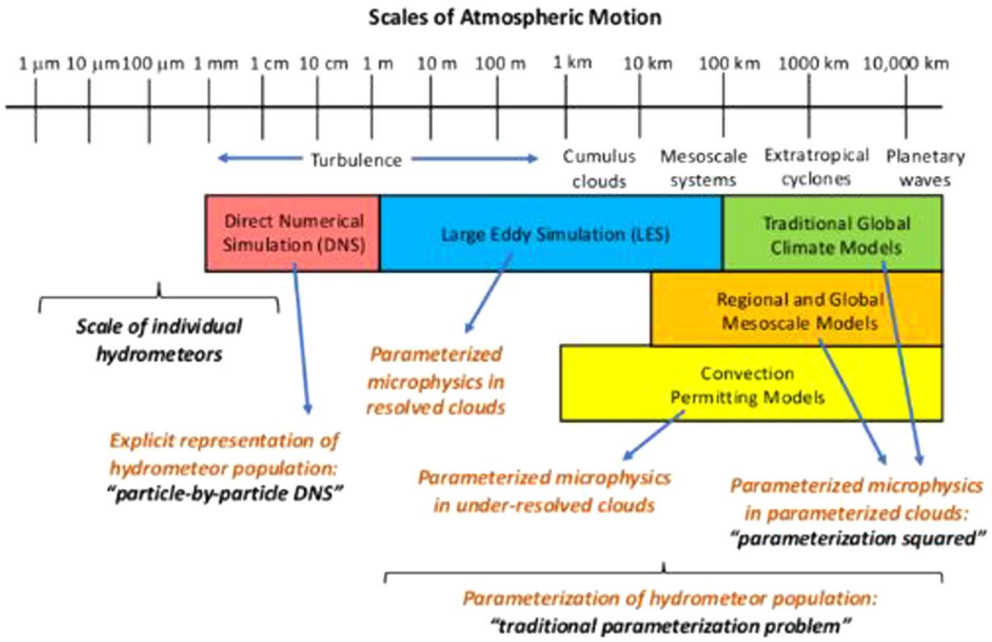


FIGURE 1.3

Schematic of the scales involved in the atmosphere and climate; Fig. 2 from Morrison, H., van Lier-Walqui, M., Fridlind, A. M., Grabowski, W. W., Harrington, J. Y., Hoose, C., et al. (2020). Confronting the challenge of modeling cloud and precipitation microphysics. Journal of Advances in Modeling Earth Systems, 12, e2019MS001689. https://doi.org/10.1029/2019MS001689, https://creativecommons.org/licenses/.

Russian), translated into English in 1965 [149]. In his book, Gandin refers to his approach as optimum rather than optimal, as we do not know the true expressions for the error variances and covariances involved. OI was the operational numerical weather prediction center's data assimilation method of choice in the 1980s and early 1990s, but because OI schemes have the restriction of linearity, and are not global solvers for the analysis, they use either a volume of observations in a local area or take only a few observations that are close to each grid point, a better approach was derived.

An alternative approach to the statistical-based OI was being developed by Yoshikazu Sasaki, where he used the idea of functionals to constrain the numerical model, given the observations. His approach would lead to the **variational methods** of data assimilation, specifically 1D, 2D, 3D, and 4DVAR. It was shown in [259] that the non-temporal variational methods, 1D, 2D, and 3DVAR, can also be derived from Bayes's equation for conditional probability. In [129] it was shown that it is also possible to describe 4DVAR as a Bayesian problem.

At the same time that Sasaki was developing his variational approach for data assimilation, Kalman was developing the **Kalman Filter**. This filter is based on control theory and it has since been shown that Kalman's approach is equivalent to an observer feedback design control system. One of the differences between the Kalman filter and the variational approach is the descriptive statistic that they are trying to find. In the variational approach we are seeking the mode of the posterior distribution—we

shall explain these terms later—while the Kalman filter seeks the minimum variance state (mean), along with the covariance matrix. For Gaussian distributions the two descriptive statistics, mean and mode, are the same.

The implementation of 4DVAR is quite expensive computationally speaking and as such the idea of including temporal information in the observations took some time to become a reality in the operational centers. It did so as a result of Courtier et al.'s 1997 paper [77] and their idea to **incrementalize** the variational approaches that enabled 4DVAR to go operational.

In the mid-1990s, the idea of using an ensemble to approximate the analysis and forecast error covariance matrix, as well as the update step, from the Kalman filter equations was presented by Evensen [111], and was called the **Ensemble Kalman Filter** (**EKF**). As a result of the 1994 paper, ensemble-based data assimilation systems have become quite widespread in their usage in the prediction of different geophysical phenomena. This led to many different versions of ensemble-based approximations to the Kalman filter, referred to as the **Ensemble Transform Kalman Filter (ETKF)**, the **Local Ensemble Transform Kalman Filter (LETKF)**, and the **Maximum Likelihood Ensemble Filter (MLEF)**, to name a few. The advantage of the ensemble-based methods is that they bring flow dependency into the analysis step, while the variational schemes assume a static model for the error covariance. Although 4DVAR does evolve the covariance matrix implicitly, it is still capturing the larger-scale errors, and not those that are referred to as **errors of the day**.

There has been movement to combine the ensemble methods with the variational methods to form **hybrid methods**. One of these approaches is the EnNDVAR, the hybrid NDVAR, which is where the **background error covariance matrix** is a weighted combination of an estimate from an ensemble of states and the static 4DVAR matrix. The other hybrid approach is NDEnVAR. The 4DVAR cost function is defined in terms of a four-dimensional trajectory which is applied as a linear model through the ensemble of trajectories.

Recently, with the need to allow for more nonlinearity, and the idea that probabilistic behavior on a smaller scale is less likely to be Gaussian, the need for data assimilation methods that allow for non-Gaussian errors has grown. One set of approaches has been derived for the lognormal distribution, and the mixed Gaussian-lognormal distribution in a variational framework, all the way to the incremental version [129,132,135–137,409]. The non-Gaussian variational approach seeks the mode of the posterior distribution, given a mixed distribution for both the background and the observation error distributions.

Another approach that has been developed to tackle the non-Gaussian aspect are methods involving **Markov Chain Monte Carlo (MCMC)** theory, which involves using an ensemble to sample the whole posterior distribution and then, from that estimation, determining the value of the descriptive statistic required. To be able to integrate this distribution in time, we then require the **particle filters** which are seen as sequential MCMC in time. Being able to model the evolution of the posterior distribution is an important approach, but the filters, if not modified, will suffer from filter degeneracy if the number of particles is not sufficiently large enough. There is a great deal of research taking place to find a way around this **curse of dimensionality**.

New to this edition of the textbook are two chapters; Lagrangian data assimilation, and artificial intelligence and data assimilation, along with new topics inside many of the existing chapter from the first edition on the theory of data assimilation.

In this book, our aim is to provide the tools to understand the mathematic, statistics, and probability theory behind the different forms of data assimilation, as well as the derivations and properties of the different schemes, so that you can decide which approach to follow. In this book we shall cover linear

algebra, random variables, descriptive statistics, univariate and multivariate distribution theory, calculus of variation, control and optimal control theory, finite differencing for initial and boundary value differential equations, semi-Lagrangian methods, the finite element model, Fourier analysis, spectral modeling, tangent linear modeling, adjoints, observations, successive correction, linear and nonlinear least squares, regression, optimum interpolation, analysis correction, variational data assimilation, physical space analysis system (PSAS) observation-space based variational data assimilation, ensemble data assimilation, Markov Chain Monte Carlo, particle filters (PF), local PF (new), particle flow filters (new), variational particle smoother (new) sigma-point Kalman filters (new), Lagrangian data assimilation (new), artificial intelligence and data assimilation (new), JEDI (new), OSE (new), OSSE (new), Green's function data assimilation (NEW) and many more new topics, and finally applications of data assimilation in different geophysical disciplines.

Therefore, at the end of this book you will hopefully have an unbiased opinion of which data assimilation approach you prefer. We have tried to be impartial, highlighting both the strengths and weaknesses of all of the data assimilation approaches. Ultimately, we would like you to understand that the goal of a data assimilation method is to:

> optimize the strengths of the models and observations, while simultaneously minimizing their weaknesses.

With this in mind, we now move on to introduce the many different mathematical and statistical disciplines that create data assimilation for the geosciences.

CHAPTER

Overview of Linear Algebra

2

Contents

The derivation of the data assimilation schemes introduced in this book require many mathematical properties, identities, and definitions for different differential operators and integral theorems. To help with these derivations we present some of the properties and techniques in this chapter as a refresher, or as an introduction to them. We start with the properties of matrices.

2.1 Properties of Matrices

Matrices play a vital role in most forms of data assimilation and the derivation of these schemes require the understanding of certain properties of the matrices. A matrix in the data assimilation literature is usually denoted as a bold capital letter, $\mathbf{A}$. The first matrix that we need to consider is the **identity** matrix which is denoted by $\mathbf{I}$. The definition of the identity matrix is

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00006-1

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}.$$

Next, a matrix $\mathbf{A}$ is said to be a square matrix if its dimensions are equal, i.e., it is of dimensions $N \times N$. The general form for an $N \times N$ matrix is given by

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1k} & \cdots & a_{1N} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2k} & \cdots & a_{2N} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3k} & \cdots & a_{3N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ a_{k1} & a_{k2} & a_{k3} & \cdots & a_{kk} & \cdots & a_{kN} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & a_{N3} & \cdots & a_{Nk} & \cdots & a_{NN} \end{pmatrix}. \tag{2.1}$$

The matrix is said to be real valued if all of its entries a_{ij} for $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, N$ are real numbers. A real $N \times N$ matrix is expressed as $\mathbf{A} \in \mathbb{R}^{N \times N}$.

2.1.1 Matrix Multiplication

The first property of matrix-matrix and matrix-vector multiplication is that it is **not** a direct element by element multiplication, although there is an element by element matrix-matrix multiplication operator which will become important in the derivation of non-Gaussian based data assimilation methods later. The rule for multiplying matrices is to multiply the ***rows by the columns*** and add the products of each element in that row-column multiplication. If we consider $\mathbf{A} \in \mathbb{R}^{2\times 2}$ and $\mathbf{B} \in \mathbb{R}^{2\times 2}$ then $\mathbf{AB} \in \mathbb{R}^{2\times 2}$ and the expression for the product is

$$\mathbf{AB} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{pmatrix}.$$

For the multiplication of two 3×3 matrices we have

$$\begin{aligned} \mathbf{AB} &= \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix} \\ &= \begin{pmatrix} \sum_{i=1}^{3} a_{1i}b_{i1} & \sum_{i=1}^{3} a_{1i}b_{i2} & \sum_{i=1}^{3} a_{1i}b_{i3} \\ \sum_{i=1}^{3} a_{2i}b_{i1} & \sum_{i=1}^{3} a_{2i}b_{i2} & \sum_{i=1}^{3} a_{2i}b_{i3} \\ \sum_{i=1}^{3} a_{3i}b_{i1} & \sum_{i=1}^{3} a_{3i}b_{i2} & \sum_{i=1}^{3} a_{3i}b_{i3} \end{pmatrix}. \end{aligned} \tag{2.2}$$

The summation expression in (2.2) is extendable to any size matrix. Matrix multiplication does not just apply to square matrices. The same formula applies for rectangular matrices so long as the number of columns of the left matrix matches the number of rows in the right matrix. If we have a 3×4 matrix and a 4×3 matrix then they are compatible for multiplication. The dimension of the matrix that arises as the product of these two matrices is 3×3. This is important because in data assimilation, we sometimes deal with matrices that are not square; as such, we need to know their dimensions correctly to ascertain the size of the problem we are solving. The rule for the dimension of the product is given by

$$\underbrace{\mathbf{A}}_{M\times N}\,\underbrace{\mathbf{B}}_{N\times Q} = \underbrace{\mathbf{AB}}_{M\times Q}.$$

The addition and subtraction of matrices is a straightforward extension of the scalar addition operator, so long as the matrices being added together are of the same dimensions. The additive operator is a direct componentwise operator. The additive matrix operators are commutative. We also have a distributive property of matrix addition and multiplication. If we have $\mathbf{A} \in \mathbb{R}^{N\times M}$, $\mathbf{B} \in \mathbb{R}^{N\times M}$, and $\mathbf{C} \in \mathbb{R}^{M\times Q}$, then

$$(\mathbf{A}+\mathbf{B})\,\mathbf{C} = \mathbf{AC}+\mathbf{BC}.$$

2.1.2 Transpose of a Matrix

The first operator acting on a matrix that we introduce is the **transpose**. The transpose of a matrix $\mathbf{A}$, is denoted as $\mathbf{A}^T$. The effect of the transpose operator is to interchange the rows and columns of the matrix. If we consider the general matrix in (2.1), then its transpose matrix is given by

$$\mathbf{A}^T = \begin{pmatrix} a_{11} & a_{21} & a_{31} & \cdots & a_{k1} & \cdots & a_{N1} \\ a_{12} & a_{22} & a_{32} & \cdots & a_{k2} & \cdots & a_{N2} \\ a_{13} & a_{23} & a_{33} & \cdots & a_{k3} & \cdots & a_{N2} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ a_{1k} & a_{2k} & a_{3k} & \cdots & a_{kk} & \cdots & a_{Nk} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ a_{1N} & a_{2N} & a_{3N} & \cdots & a_{kN} & \cdots & a_{NN} \end{pmatrix}. \tag{2.3}$$

A special class of matrices that are important in data assimilation are **symmetric** matrices. A square matrix is said to be symmetric if $a_{ij} = a_{ji}$ for $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, N$, for $i \neq j$. An important property of symmetric matrices is that $\mathbf{A}^T = \mathbf{A}$.

Exercise 2.1. *Find the transpose of the following matrices, identifying which, if any, are symmetric:*

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \\ 3 & 6 & 10 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & -5 & 2 & 0 \\ -5 & 0 & 3 & 0 \\ 2 & 3 & -3.5 & 7 \\ 2 & 0 & 3 & 0 \end{pmatrix}.$$

2.1.3 Determinants of Matrices

An important feature of matrices is their determinant. The determinant can either be denoted as $|\mathbf{A}|$ or $\det(\mathbf{A})$. The determinant is only applicable to square matrices. We start by considering a general 2×2

matrix's determinant, which is defined by

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{21}a_{22}.$$

For a general 3×3 matrix, the technique to derive the determinant involves the expansion as follows

$$\begin{aligned} \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} &= a_{11} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix}, \\ &= a_{11}\left(a_{21}a_{32} - a_{31}a_{22}\right) - a_{12}\left(a_{21}a_{33} - a_{31}a_{23}\right) \\ &\quad + a_{13}\left(a_{21}a_{32} - a_{31}a_{22}\right). \end{aligned} \tag{2.4}$$

Finally we consider the general 4×4 case which is

$$\begin{aligned} \begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix} &= a_{11} \begin{vmatrix} a_{22} & a_{23} & a_{24} \\ a_{32} & a_{33} & a_{34} \\ a_{42} & a_{43} & a_{44} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} & a_{24} \\ a_{31} & a_{33} & a_{34} \\ a_{41} & a_{43} & a_{44} \end{vmatrix} \\ &\quad + a_{13} \begin{vmatrix} a_{21} & a_{22} & a_{24} \\ a_{31} & a_{32} & a_{34} \\ a_{41} & a_{42} & a_{44} \end{vmatrix} - a_{14} \begin{vmatrix} a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \\ a_{41} & a_{42} & a_{43} \end{vmatrix}, \end{aligned} \tag{2.5}$$

where the next set of determinants in (2.4) and (2.5) are referred to as the *minors* of $\mathbf{A}$ and are expanded as demonstrated for the 3×3 case. As can be seen in (2.4) and (2.5), the sign of the factors multiplying the minors are alternating. The signs for the specific elements in a 4×4 matrix are

$$\begin{pmatrix} + & - & + & - \\ - & + & - & + \\ + & - & + & - \\ - & + & - & + \end{pmatrix}. \tag{2.6}$$

It should be noted that it does not matter which row or column you expand the determinant about; you obtain the same answer. This is important for saving time where there are zeros present in any line or column of a matrix, as this removes associated minors from having to be evaluated.

Example 2.2. *Find the determinants of the following three matrices:*

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 10 & 1000 & 2 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 2 & 3 & 0 & 1 \end{pmatrix}. \tag{2.7}$$

Solution. Taking $\mathbf{A}$ first we have

$$|\mathbf{A}| = (1 \times 4) - (2 \times 2) = 0.$$

For $\mathbf{B}$, expanding the first row yields

$$|\mathbf{B}| = 1 \begin{vmatrix} 5 & 6 \\ 8 & 9 \end{vmatrix} - 2 \begin{vmatrix} 4 & 6 \\ 7 & 9 \end{vmatrix} + 3 \begin{vmatrix} 4 & 5 \\ 7 & 8 \end{vmatrix}$$

$$= 1 \times ((5 \times 9) - (8 \times 6)) - 2 \times ((4 \times 9) - (7 \times 6)) + 3 \times ((4 \times 8) - (7 \times 5))$$
$$= -3 + 12 - 9 = 0.$$

For matrix $\mathbf{C}$ we are going to expand the determinant about the second row. The reason for this is that there are three zeros on this row which means that there is only one minor that needs to be evaluated. Therefore, the first step in finding $|\mathbf{C}|$ is

$$|\mathbf{C}| = 1 \begin{vmatrix} 1 & 10 & 1000 \\ 0 & 1 & 0 \\ 2 & 3 & 0 \end{vmatrix}.$$

Next expanding about the second row in the remaining minor gives

$$\begin{vmatrix} 1 & 10 & 1000 \\ 0 & 1 & 0 \\ 2 & 3 & 0 \end{vmatrix} = -1 \begin{vmatrix} 1 & 1000 \\ 2 & 0 \end{vmatrix}$$
$$= -1 \times (-2000) = 2000.$$

Note: The plus and minus signs for minor reset for each subdeterminant.

There are many important properties of determinants that play vital roles in the derivation of numerous parts involved in data assimilation. We start by assuming that the two matrices $\mathbf{A} \in \mathbb{R}^{N \times N}$ and $\mathbf{B} \in \mathbb{R}^{N \times N}$ are real and square matrices, then

1. $|\mathbf{AB}| = |\mathbf{A}|\,|\mathbf{B}|$,
2. $\left|\mathbf{A}^{-1}\right| = \frac{1}{|\mathbf{A}|} = |\mathbf{A}|^{-1}$,
3. $\left|\mathbf{A}^{T}\right| = |\mathbf{A}|$,
4. $|\mathbf{I}| = 1$, and
5. $|c\mathbf{A}| = c^{N}\,|\mathbf{A}|$.

The second property above is referring to the inverse of the matrix which is defined in the next subsection.

2.1.4 Inversions of Matrices

Matrix inverses play a very important role in many forms of data assimilation. The inversion of a matrix is not a trivial operation to perform as the dimensions of the matrices become large. As with the determinants we start with the general 2×2 matrix, where the inverse of $\mathbf{A}$ is defined by

$$\mathbf{A}^{-1} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}^{-1} \equiv \frac{1}{a_{11}a_{22} - a_{21}a_{12}} \begin{pmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{pmatrix}. \tag{2.8}$$

The denominator of the fraction multiplying the inverse matrix in (2.8) is the determinant of $\mathbf{A}$. The rule for a 2×2 matrix is to interchange the diagonal entries and take the negative of the two off diagonal entries. However, as can be seen in (2.8), the inverse of a matrix can only exist if $\det(\mathbf{A}) \neq 0$. If a matrix does have a determinant equal to zero, then it is said to be **singular**, or non-invertible.

For matrices larger than order 2, the associated inversion become quite cumbersome. Before we consider higher dimensional matrices we first introduce the matrix of cofactors. We shall illustrate this

for a 3×3 matrix, but the definitions expand to larger dimension matrices. The matrix of cofactors is defined as

$$\mathbf{CO}(\mathbf{A}) = \begin{pmatrix} C_{11} & -C_{12} & C_{13} \\ -C_{21} & C_{22} & -C_{23} \\ C_{31} & -C_{32} & C_{33} \end{pmatrix}, \tag{2.9}$$

where the C_{ij} are the minors expanded about the location of the index ij. **Note:** The cofactors follow the plus and minus signs as presented in (2.6). For the 3×3 case the cofactors are

$$\begin{aligned} C_{11} &= a_{22}a_{33} - a_{32}a_{23}, & C_{12} &= a_{21}a_{33} - a_{31}a_{23}, & C_{13} &= a_{21}a_{32} - a_{31}a_{22}, \\ C_{21} &= a_{12}a_{33} - a_{32}a_{13}, & C_{22} &= a_{11}a_{33} - a_{31}a_{13}, & C_{23} &= a_{11}a_{32} - a_{31}a_{12}, \\ C_{31} &= a_{12}a_{23} - a_{22}a_{13}, & C_{32} &= a_{11}a_{23} - a_{21}a_{13}, & C_{33} &= a_{11}a_{22} - a_{21}a_{12}. \end{aligned}$$

Finally the definition for the inverse of a square matrix is

$$\mathbf{A}^{-1} = \frac{1}{\det(\mathbf{A})}\mathbf{CO}(\mathbf{A})^T. \tag{2.10}$$

For the general 3×3 matrix the general expression for the inverse is

$$\mathbf{A}^{-1} \equiv \frac{1}{|\mathbf{A}|}\begin{pmatrix} a_{22}a_{33} - a_{32}a_{23} & a_{32}a_{13} - a_{12}a_{33} & a_{12}a_{23} - a_{22}a_{13} \\ a_{31}a_{23} - a_{21}a_{33} & a_{11}a_{33} - a_{31}a_{13} & a_{21}a_{13} - a_{11}a_{23} \\ a_{21}a_{32} - a_{31}a_{22} & a_{31}a_{12} - a_{11}a_{32} & a_{11}a_{22} - a_{21}a_{12} \end{pmatrix}, \tag{2.11}$$

where the negative cofactors have interchanged the minus signs of the sum-matrices' determinants.

Example 2.3. *Find the inverse of the following matrix,*

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \\ 4 & 5 & 5 \end{pmatrix}. \tag{2.12}$$

The first step is to find the determinant of $\mathbf{A}$. Expanding about by the first row we have

$$\begin{aligned} |\mathbf{A}| &= 1((2 \times 5) - (5 \times 1)) - 2((3 \times 5) - (4 \times 1)) + 3((3 \times 5) - (4 \times 2)), \\ &= 1(10 - 5) - 2(15 - 4) + 3(15 - 8), \\ &= 5 - 22 + 21, \\ &= 4. \end{aligned}$$

Next we form the matrix of cofactors which can easily be shown for this example to be

$$\mathbf{CO}(\mathbf{A}) = \begin{pmatrix} 5 & -11 & 7 \\ 5 & -7 & 3 \\ -4 & 8 & -4 \end{pmatrix}.$$

Therefore, the inverse of the matrix in (2.12) is

$$\mathbf{A}^{-1} = \frac{1}{4}\begin{pmatrix} 5 & 5 & -4 \\ -11 & -7 & 8 \\ 7 & 3 & -4 \end{pmatrix}. \tag{2.13}$$

An important equation that links the matrix inverse to the identity matrix is

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{A}^{-1}\mathbf{A} = \mathbf{I}. \tag{2.14}$$

Exercise 2.4. *Verify that the matrix in* (2.13) *is the inverse of the matrix in* (2.12).
Exercise 2.5. *Identify which of the following matrices are singular by finding their determinants:*

$$\mathbf{A} = \begin{pmatrix} -2 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 4 & 1 & 1 \\ 1 & 3 & 1 \\ 2 & 5 & 7 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 10 & 0 & 0 \end{pmatrix}.$$

Now that we have introduced the transpose and the inverse of a matrix we consider important properties of these operators on the product of matrices, which again will play an important part in the derivations of many of the equations that are involved in different aspects of data assimilation.

We first consider the inverse and the transpose of the product of two matrices. These important properties are:

$$(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}, \tag{2.15a}$$

$$(\mathbf{A}\mathbf{B})^{T} = \mathbf{B}^{T}\mathbf{A}^{T}. \tag{2.15b}$$

Therefore, the inverse of the product is the product of the inverses in reverse order. This is also true for the transpose as well. Note that for the transpose operator the matrices do not have to be square matrices, however, for the inverse operator the matrices do have to be square. The proof for (2.15a) is quite straightforward and is given below.

Proof. We start with the relationship $(\mathbf{A}\mathbf{B})^{-1}\mathbf{A}\mathbf{B} = \mathbf{I}$. Multiplying on the right of both sides by $\mathbf{B}^{-1}$ gives $(\mathbf{A}\mathbf{B})^{-1}\mathbf{A} = \mathbf{B}^{-1}$. Now multiplying on the right of both sides by $\mathbf{A}^{-1}$ results in $(\mathbf{A}\mathbf{B})^{-1} = \mathbf{B}^{-1}\mathbf{A}^{-1}$.

The property of the inverse of the product of two matrices can be extended to the product of n matrices, i.e.,

$$(\mathbf{A}_1\mathbf{A}_2\cdots\mathbf{A}_n)^{-1} = \mathbf{A}_n^{-1}\mathbf{A}_{n-1}^{-1}\cdots\mathbf{A}_1^{-1}.$$

The same is also true for the transpose operator as well

$$(\mathbf{A}_1\mathbf{A}_2\cdots\mathbf{A}_n)^{T} = \mathbf{A}_n^{T}\mathbf{A}_{n-1}^{T}\cdots\mathbf{A}_1^{T}.$$

An important property that links the transpose and the inverse is that the order that you perform the operators can be interchanged, i.e.,

$$\left((\mathbf{A})^{T}\right)^{-1} \equiv \left((\mathbf{A})^{-1}\right)^{T} = \mathbf{A}^{-T}.$$

2.1.5 Rank, Linear Independence and Dependence

As we saw in the previous subsection there were matrices that were not invertible and had determinants equal to zero. The reason for this is that these matrices had either rows or columns that were equal to the sum of all or some of the other columns or rows.

A method to determine if a square matrix is singular is referred to as row or column reduction. The row reduction technique, which is the same for column reduction, is to take the leading diagonal that does not have all zeros below it and see if any rows have a 1 in that column. If not, then divide that row by the diagonal entry to make the leading diagonal a 1. The next step is to remove the entries below the diagonal entry. This is achieved through multiplying the leading diagonal entry by the factor multiplying the entries below them, and then subtracting this scaled row from the matching row below. This process is repeated for each diagonal entry. If rows or columns of zeros occur then the total number of non-zero rows of columns is referred to as the **rank** of the matrix. The remaining rows or columns are referred to as being **linearly independent**, while the rows or columns that are all zeros are referred to as being **linearly dependent**. Matrices that have a rank equal to the dimension of that matrix are said to be **full-rank**, while those who's rank is less than the dimensions of the matrix are said to be **rank-deficient**.

Example 2.6. *Consider the matrix* $\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$, *show that it has rank* $(\mathbf{A}) = 2$.

Proof. We start by noticing that we have a 1 on the leading diagonal. Therefore we can begin by eliminating the 4 and the 7 below by subtracting 4r1 from r2 and 7r1 from r3. This leaves $\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -12 \end{pmatrix}$. Now it is clear why the $\det(\mathbf{A}) = 0$ earlier. It is now possible to remove the last row through r3–2r2. Therefore, the final form is $\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & 0 \end{pmatrix}$. This then implies that there are two linearly independent rows, so the matrix has rank $= 2$.

Knowing about linear independence is a critical tool in diagnosing mistakes in matrix coding and formulations. If you know that your matrix should be invertible, but it appears not to be, then either it is a coding problem where two or more rows have been repeated, which makes the matrix singular, or the formulation of the problem is referred to as **ill-posed**. We shall explore ill-posedness in Chapter 8.

When row-reducing a rectangular matrix, the number of linearly independent columns is referred to as the matrix's **column-rank**, and the number of linearly independent rows is referred to as the **row-rank**.

2.1.6 Matrix Structures

The first matrix structure that has explicitly been identified is symmetry. The identity matrix introduced in Section 2.1 is a diagonal matrix. The diagonal matrices have the property that their inverses are simply the reciprocal of the diagonal entries. Diagonal matrices play important roles in data assimilation, primarily due to the ease of finding their inverses. However, if it is not possible to obtain a diagonal matrix, the next best matrix form is a banded diagonal. The 4×4 example in (2.16) is a specific type of banded matrix referred to as a **tri-diagonal** matrix,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{pmatrix}. \tag{2.16}$$

Matrices similar to that in (2.16) occur in finite differences approximations to certain type of differential equations. Matrices that have entries either side of the diagonal entry, but not both sides, and the first diagonal above or below are referred to as **bi-diagonal** matrices.

Another form of diagonal matrix is the **block diagonal**. An example of a block diagonal matrix is

$$\mathbf{F} = \begin{pmatrix} \mathbf{A} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D} \end{pmatrix}, \tag{2.17}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{m\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times p}$, $\mathbf{D} \in \mathbb{R}^{q\times q}$, and the $\mathbf{0}$ represent the part of the matrix that have only zeros there. The dimensions of $\mathbf{F}$ are $(n+m+p+q)\times(n+m+p+q)$. Note that the four matrices on the diagonal may still be full and difficult to invert. However, the inverse of $\mathbf{F}$ is

$$\mathbf{F}^{-1} = \begin{pmatrix} \mathbf{A}^{-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}^{-1} \end{pmatrix}. \tag{2.18}$$

Block diagonal matrices can occur in data assimilation when trying to decouple/decorrelate certain variables to make their covariance matrix less full, as well as making it both manageable for computational storage and easier inversion.

The next set of matrix structures are the **triangular** forms. There are two forms of this class: the lower triangula, $\mathbf{L}$, and the upper triangular, $\mathbf{U}$, and in general forms for both given by

$$\mathbf{L} = \begin{pmatrix} L_{11} & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} \end{pmatrix}, \quad \mathbf{U} = \begin{pmatrix} U_{11} & U_{12} & U_{13} & U_{14} \\ 0 & U_{22} & U_{23} & U_{24} \\ 0 & 0 & U_{33} & U_{34} \\ 0 & 0 & 0 & U_{44} \end{pmatrix}. \tag{2.19}$$

These occur in decompositions of matrices that are often associated with the solver of large set of simultaneous equations. We shall go into more detail about this specific decomposition in Section 2.4.

Another useful class of matrices are **orthogonal matrices**. These have the important property that $\mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A} = \mathbf{I}$. These matrices can occur when different transforms are applied to certain variables to enable the problem to be easier to invert.

2.2 Matrix and Vector Norms

Norms play an important part in determining not only the performance of the data assimilation algorithm, but also the error analysis. The norm can be used to provide the bounds of the accuracy of the performance of approximations that have been made in the assimilation schemes. The first set of norms that are consider here are the vector norms.

2.2.1 Vector Norms

The purpose of a vector norm is to provide a measure, of some form, of a specific vector. The definitions of the vector norms apply to both vectors in the real number and complex number spaces. The mathematical definition of a vector norm is given by the following.

Definition 2.7. A norm of the vector, $\mathbf{x} \in \mathbb{R}^n$ or $\mathbb{C}^n$, is denoted by $||\mathbf{x}||$ and is a mapping from $\mathbb{R}^n$, or $\mathbb{C}^n$, to the space of real numbers, $\mathbb{R}$, that have the following properties:

1. $||\mathbf{x}|| \geq 0$ with equality only occurring for $\mathbf{x} = \mathbf{0}$.
2. For any scalar, λ, then $||\lambda \mathbf{x}|| = |\lambda| \cdot ||\mathbf{x}||$.
3. For any vectors $\mathbf{x}$ and $\mathbf{y}$, then $||\mathbf{x} + \mathbf{y}|| \leq ||\mathbf{x}|| + ||\mathbf{y}||$. This inequality of the sums is referred to as the **triangle inequality**.

A class of norms are referred to as the l_p norms, denoted by $||\mathbf{x}||_p$, are defined by

$$||\mathbf{x}||_p \equiv \left(\sum_{i=1}^{n} |\mathbf{x}|^p \right)^{1/p}.$$

If we consider the case where $p = 2$, which is often referred to in the data assimilation literature as the l_2 norm, then the associated definition for this norm is given by

$$||\mathbf{x}||_2 \equiv \left(\sum_{i=1}^{n} |\mathbf{x}|^2 \right)^{1/2} = |\mathbf{x}|, \tag{2.20}$$

and is the normal length.

There are three l_p norms that are commonly used. These are as follows:

1. **Euclidean Norm** ($p = 2$),

$$||\mathbf{x}||_2 \equiv \left(\sum_{i=1}^{n} |\mathbf{x}|^2 \right)^{1/2} = \left(\bar{\mathbf{x}}^T \mathbf{x} \right)^{1/2}, \tag{2.21}$$

where the overhead bar is the complex conjugate of the vector $\mathbf{x}$.

2. **Absolute Norm** ($p = 1$),

$$||\mathbf{x}||_1 \equiv \sum_{i=1}^{n} |\mathbf{x}|. \tag{2.22}$$

3. **Maximum Norm** l_∞, this norm is also referred to as the infinity norm,

$$||\mathbf{x}||_\infty = \max |\mathbf{x}|, \quad 1 \leq i \leq n. \tag{2.23}$$

Example 2.8. *Find the Euclidean norm, the absolute norm and the infinity norm of the following two vectors:*

$$\boldsymbol{x} = \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix} \quad \textit{and} \quad \boldsymbol{y} = \begin{pmatrix} 1 \\ 1+i \end{pmatrix}. \tag{2.24}$$

Taking the $\mathbf{x}$ vector first, then $||\mathbf{x}||_2 = \sqrt{14}$, $||\mathbf{x}||_1 = 6$, and $||\mathbf{x}||_\infty = 3$. For the $\mathbf{y}$ vector we have $||\mathbf{y}||_2 = \begin{pmatrix} 1 & 1+i \end{pmatrix} \begin{pmatrix} 1 \\ 1+i \end{pmatrix} = \sqrt{3}$, $||\mathbf{y}||_1 = 1 + \sqrt{2}$, and $||\mathbf{y}||_\infty = |1+i| = \sqrt{2}$.

Exercise 2.9. *Find the Euclidean, maximum and absolute norms for the following vectors*

$$\boldsymbol{x} = \begin{pmatrix} 2 \\ 3 \\ -4 \\ 1 \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} -4 \\ 6 \\ -2.5 \\ -6.3 \end{pmatrix}, \quad \boldsymbol{z} = \begin{pmatrix} -1.6 \\ 7 \\ -4.8 \\ 3 \end{pmatrix},$$

and verify the triangle identity for the Euclidean norm for $\boldsymbol{x}+\boldsymbol{y}$, $\boldsymbol{x}+\boldsymbol{z}$, *and* $\boldsymbol{y}+\boldsymbol{z}$.

Another useful property of norms is that of *equivalence* between the vector norm. All norms that are operating on $\mathbb{R}^n$ are **equivalent** such that given two positive constants c_1 and c_2, and two different norms $||\cdot||_\alpha$ and $||\cdot||_\beta$, then

$$c_1 ||\mathbf{x}||_\alpha \leq ||\mathbf{x}||_\beta \leq c_2 ||\mathbf{x}||_\alpha .$$

It can easily be shown that the necessary constants to show equivalence between the three p norms above are

$$\begin{aligned} ||\mathbf{x}||_2 &\leq ||\mathbf{x}||_1 \leq \sqrt{n}\, ||\mathbf{x}||_2 , \\ ||\mathbf{x}||_\infty &\leq ||\mathbf{x}||_2 \leq \sqrt{n}\, ||\mathbf{x}||_\infty , \\ ||\mathbf{x}||_\infty &\leq ||\mathbf{x}||_1 \leq n\, ||\mathbf{x}||_\infty . \end{aligned}$$

2.2.2 Matrix Norms

The norm of a real $n \times n$ matrix in the $\mathbb{R}^{n\times n}$ space, or a complex numbered $n \times n$ matrix in the $\mathbb{C}^{n\times n}$, is a mapping from $\mathbb{R}^{n\times n}$ or $\mathbb{C}^{n\times n}$ to $\mathbb{R}$ and satisfies the following properties:

1. $||\mathbf{A}|| > 0, \quad \mathbf{A} \neq 0, \quad ||\mathbf{A}|| = 0 \leftrightarrow \mathbf{A} = 0.$
2. $||\lambda \mathbf{A}|| = |\lambda|\, ||\mathbf{A}||$ where λ is a scalar and $\lambda \in \mathbb{R}$ or $\lambda \in \mathbb{C}$.
3. $||\mathbf{A} + \mathbf{B}|| \leq ||\mathbf{A}|| + ||\mathbf{B}||$ (Triangle inequality).
4. $||\mathbf{AB}|| \leq ||\mathbf{A}||\, ||\mathbf{B}||$.

We now introduce a definition to link the matrix norms to vector norms.

Definition 2.10. If a matrix norm and a vector norm satisfy

$$||\mathbf{Ax}|| \leq ||\mathbf{A}||\, ||\mathbf{x}|| , \qquad \begin{matrix} \forall \mathbf{A} \in \mathbb{R}^{n\times n}, \mathbb{C}^{n\times n}, \\ \forall \mathbf{x} \in \mathbb{R}^n, \mathbb{C}^n, \end{matrix} \tag{2.25}$$

then the matrix norm is said to be **consistent** with the vector norm.

It should be noted that the norm of the left-hand side of the inequality in (2.25) is a vector norm, and that the right-hand side of the inequality in (2.25) is the product of a matrix and a vector norm.

Definition 2.11. Given some vector norms, we define a matrix norm as follows:

$$||\mathbf{A}|| = \max_{\mathbf{x}\neq\mathbf{0}} \frac{||\mathbf{Ax}||}{||\mathbf{x}||}. \tag{2.26}$$

The matrix norm $||\cdot||$ is called a **subordinate** matrix norm and is always **consistent** with the vector norm.

Given Definitions 2.10 and 2.11, the subordinate l_p norms for the maximum/infinity, the absolute and the l_2 vector norms are

1. $||\mathbf{A}||_\infty = \max_i \sum_{j=1}^{n} |a_{i,j}|$, which is the maximum of the row sums.
2. $||\mathbf{A}||_1 = \max_j \sum_{j=1}^{n} |a_{i,j}|$, which is the maximum of the column sums.
3. $||\mathbf{A}||_2 = \max_i \left|\lambda_i \left(\mathbf{A}^T\mathbf{A}\right)\right|^{1/2}$, where $\mathbf{A} \in \mathbb{R}^{n\times n}$ and λ_i is the eigenvalue of $\mathbf{A}^T\mathbf{A}$. For the complex number valued matrices, i.e., $\mathbf{A} \in \mathbb{C}^{n\times n}$ then the Hermitian transpose, which is defined as $\mathbf{A}^H \equiv \overline{\mathbf{A}}^T$, where the entries in $\overline{\mathbf{A}} = \overline{a}_{i,j}$ are the complex conjugates and the definition of the 2-norm for complex number valued matrices is $||\mathbf{A}||_2 = \max_i \left|\lambda_i \left(\mathbf{A}^H\mathbf{A}\right)\right|^{1/2}$.

Note: All of the matrix norms above are consistent with the corresponding l_p vector norm.

However, the l_2 matrix norm defined above appears quite different from its consistent vector norm. There is one more vector norm that we shall introduce here and appears to be similar in structure to the l_2 vector norm and is referred to as the **Frobenius norm**, which is defined by

$$||\mathbf{A}||_F \equiv \sqrt{\left(\sum_{i=1}^{n}\sum_{j=1}^{n} |a_{i,j}|^2\right)}. \tag{2.27}$$

The Frobenius norm is consistent with the l_2 vector norm such that $||\mathbf{Ax}||_2 \leq ||\mathbf{A}||_F\, ||\mathbf{x}||_2$; however, it should be noted that the Frobenius norm is not subordinate to any vector norm.

2.2.3 Conditioning of Matrices

Conditioning of matrices play a vital part in many parts of geophysical numerical modeling. In this subsection the condition number of a matrix is presented and derived, as well as an explanation on how to interrupt the number as well as its implication of the accuracy of the numerical approximation that we are applying. If the problem we are approximating is sensitive to small computational errors then we need a measure to quantify this sensitivity. Consider the generalized case

$$F(x, y) = 0, \tag{2.28}$$

where we are seeking the unknown x, and have y that contains data that the solution depends upon.

The problem expressed in (2.28) is said to be well-posed, or stable depending on the literature, if the solution x depends in a continuous way on y. This is equivalent to saying that if you have a sequence $\{y_n\}$ that is tending towards y, then the corresponding sequence for x, $\{x_n\}$ must also approach x in some way. As we are interested in sensitivity to small changes, an equivalent definition of well-posedness is to consider that if there are small changes to y, then there are only small changes in x. Problems that do not satisfy these loose definitions are referred to as **ill-posed or unstable**.

If a problem is ill-posed, then there are serious implication on the ability to solve these types of problems. However, a continuous problem may be stable, but the associated numerical approximations could encounter difficulties when solving for a solution. The condition number is a measure to ascertain how stable a problem is.

So why is the condition number so important? The condition number attempts to give guidance, or a measure, of the worst possible effect on the solution x, given small perturbations to y. To obtain the definition for the condition number, we consider small perturbations to both x and y, denoted by δx and δy respectively, implying that $x + \delta x$ the solution to the perturbed version of (2.28), which is expressed as

$$F(x + \delta x, y + \delta y) = 0. \tag{2.29}$$

Given (2.29), the condition number for (2.28), denoted as $\kappa(x)$, is defined as

$$\kappa(x) = \underbrace{\text{Supremum}}_{\delta y} \frac{||\delta x|| / ||x||}{||\delta y|| / ||y||}. \tag{2.30}$$

If we were solving for a vector then the measure in (2.30) would be one of the vector norms defined in (2.21)–(2.23); however, as a caveat it should be noted that a different norm may be needed for x and y. In [17], the supremum is defined as the largest possible value for δy such that the perturbed equation (2.30) "*makes sense*."

In [17], the way to interpret (2.30) is explained as a measure of the sensitivity of solution x to small changes in the data. Also in [17] is an explanation of how to understand what the magnitude of κ means. If we consider the case when $\kappa(x) = 10^{10}$, and given that δy is assumed to be quite small, then this implies that δx must be quite large, which means that the problem that we are solving is very sensitive to small changes in the data. However, if $\kappa(x) \leq 10$, then small changes in the data, δy, result in small changes in x.

Therefore, the condition number informs us whether or not the continuous problem that we are going to approximate is sensitive to small changes. When numerically approximating continuous problems, it is hoped that the most accurate approximating is used so that the errors introduced from the scheme will not result in large changes in the solution.

However, it is not always possible to calculate the condition number of the continuous problem that you are seeking solutions to, therefore, there is a different condition number associated with matrices that occurs in the numerical approximation to the continuous problem that can also give guidance about the order of accuracy to expect in the solution.

2.2.4 Matrix Condition Number

The matrix condition number come about through considering the error analysis of the matrix equation

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{2.31}$$

The error analysis is based upon determining the sensitivity of the solution $\mathbf{x}$ to small perturbations. This starts by considering the perturbed matrix equation

$$\mathbf{A}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{A} + \mathbf{r}, \tag{2.32}$$

where $\mathbf{r}$ is the **residual**. We also make the assumption that (2.31) has a unique solution, $\mathbf{x}$, of order n.

Subtracting (2.31) from (2.32) results in

$$\mathbf{A}\delta\mathbf{x} = \mathbf{r} \Rightarrow \delta\mathbf{x} = \mathbf{A}^{-1}\mathbf{r}. \tag{2.33}$$

Recalling that the definition for continuous problem's condition number is the ratio of the norm of the perturbed solution to the norm of the true solution to the ratio of the norm of the perturbed data to the norm of the true data, implies for (2.31) and (2.32), that the expression just mentioned is equivalent to

$$\frac{\frac{||\delta \mathbf{x}||}{||\mathbf{x}||}}{\frac{||\mathbf{r}||}{||\mathbf{b}||}}, \tag{2.34}$$

where the expression in (2.34) is a way to examine the stability of (2.31) for all perturbations $\mathbf{r}$ in $\mathbb{R}^n$ that are small relative to $\mathbf{b}$.

To obtain the expression in (2.34) we first take the norm of the equations in (2.33), which results in

$$||\mathbf{r}|| \leq ||\mathbf{A}||\,||\delta \mathbf{x}||\,, \quad ||\delta \mathbf{x}|| \leq \left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right| ||\mathbf{r}||\,. \tag{2.35}$$

The next step is to divide the first inequality in (2.35) by $||\mathbf{A}||\,||\mathbf{x}||$, and divide the second inequality in (2.35) by $||\mathbf{x}||$, which enables us to find an inequality bound for the ratio of the norm of $\delta \mathbf{x}$ to $\mathbf{x}$ as

$$\frac{||\mathbf{r}||}{||\mathbf{A}||\,||\mathbf{x}||} \leq \frac{||\delta \mathbf{x}||}{||\mathbf{x}||} \leq \frac{\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right| ||\mathbf{r}||}{||\mathbf{x}||}. \tag{2.36}$$

The next step in the derivation require that the matrix norm is induced by the vector norm which implies the following two inequalities

$$||\mathbf{b}|| \leq ||\mathbf{A}||\,||\mathbf{x}||\,, \quad \text{and} \quad ||\mathbf{x}|| \leq \left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right| ||\mathbf{b}||\,. \tag{2.37}$$

Substituting the two inequalities in (2.37) into the inequality in (2.36) results in

$$\frac{1}{||\mathbf{A}||\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right|} \frac{||\mathbf{r}||}{||\mathbf{b}||} \leq \frac{||\delta \mathbf{x}||}{||\mathbf{x}||} \leq ||\mathbf{A}||\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right| \frac{||\mathbf{r}||}{||\mathbf{b}||}. \tag{2.38}$$

Dividing throughout (2.38) by $\frac{||\mathbf{r}||}{||\mathbf{b}||}$ results in the expression that we wish to bound in the center of the inequality and the expression for the upper bound of $||\mathbf{A}||\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right|$. It is the product of the norm of the matrix multiplying the norm of the inverse of the matrix that is referred to as the condition number. Therefore,

$$\text{cond}\,(\mathbf{A}) \equiv \kappa\,(\mathbf{A}) \equiv ||\mathbf{A}||\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right|. \tag{2.39}$$

Given the definition for the condition number, the next step is to understand how to interrupt the meaning of the number and the guidance that it gives towards the accuracy of the solution to the matrix equation. A property to notice here is that the condition number will be dependent on the norm that is chosen. However, as shown below, the lower bound for the condition number, no matter the choice of norm, is always 1, as

$$1 \leq ||\mathbf{I}|| \leq \left|\left|\mathbf{A}\mathbf{A}^{-\mathbf{1}}\right|\right| \leq ||\mathbf{A}||\left|\left|\mathbf{A}^{-\mathbf{1}}\right|\right| = \kappa\,(\mathbf{A})\,. \tag{2.40}$$

Given the expression in (2.40), it is clear that if the condition number is close to 1, then we can see from (2.39) that relatively small perturbations in $\mathbf{b}$ lead to near similar relatively small perturbations in

x. However, the opposite is true that if the condition number is large then relatively small perturbations to **b** leads to large changes in **x**.

Example 2.12. *Consider the following matrix equation which represents a numerical approximation to advection on a periodic domain. How conditioned is this numerical problem?*

$$\begin{pmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{pmatrix}. \tag{2.41}$$

Upon looking at the matrix in (2.41), it is quite clear that this matrix is singular and has a condition number of ∞ as a result of the zero determinant. Due to the condition number being so large, it can be implied that the continuous problem that this discrete approximation represents is ill-posed. There are techniques to deal with these types of problems. One is to add a small number to the diagonal entry to perturb the matrix away from singularity. Another approach is to fix a point, which is equivalent to re-writing the problem with an extra constraint and then discretizing as before for the remaining points. We shall go into more detail about the actual model (2.41) arises from in Chapter 8.

2.3 Eigenvalues and Eigenvectors

As with many properties of matrices and vectors that have been introduced so far, eigenvalues and eigenvectors also play an important part in many aspects of numerical modeling, matrix decompositions, control theory, covariance modeling, and preconditioning to name but a few areas. The **eigenvalues** of a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ are the roots of the associated **characteristic polynomial**, which is denoted by $p(z) = \det(z\mathbf{I} - \mathbf{A})$. The collection of the roots, eigenvalues, is called the **spectrum** of $\mathbf{A}$ and is denoted by $\lambda(\mathbf{A}) = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$, where the λ_is are the eigenvalues. An important property of eigenvalues is that the determinant of a square matrix is equal to the product of its eigenvalues,

$$\det(\mathbf{A}) = \lambda_1 \lambda_2 \ldots \lambda_n.$$

Eigenvalues are also related to the trace of a matrix, which is the sum of a matrix's diagonal entries, given by

$$\operatorname{tr}(\mathbf{a}) = \sum_{i=1}^{n} a_{ii}.$$

The relationship of eigenvalues to the trace of a matrix is defined as $\operatorname{tr}(\mathbf{A}) = \lambda_1 + \lambda_2 + \cdots + \lambda_n$.

If the eigenvalue λ is in the spectrum of $\mathbf{A}$, i.e., $\lambda \in \lambda(\mathbf{A})$, then the non-zero vector $\mathbf{x} \in \mathbb{C}^n$ that satisfies the equation

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \tag{2.42}$$

is referred to as its **eigenvector**. There are two types of eigenvectors; the first are the right eigenvectors which satisfy (2.42). The second set are called the left eigenvectors that satisfy the equation

$$\mathbf{x}^H\mathbf{A} = \lambda\mathbf{x}^H, \tag{2.43}$$

where the superscript H is the **conjugate transpose**, which if $\mathbf{x} \in \mathbb{R}^n$ then this is the transposed defined earlier. Unless stated otherwise, when we use the term "eigenvector" we are referring to the right eigenvectors. The term "conjugate in complex number" refers to a pair of imaginary numbers that have equal real parts, and also have equal imaginary parts in magnitude, but are of opposite sign, i.e., $1 - 2i$ is the conjugate of $1 + 2i$.

Example 2.13. *Find the eigenvalues and eigenvectors of the following* 2×2 *real matrix* $\mathbf{A}$ *given by*

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 3 & 1 \end{pmatrix}.$$

The first step in finding the eigenvalue is to form the matrix $\mathbf{A} - \lambda\mathbf{I}$, which for this example is

$$\mathbf{A} - \lambda\mathbf{I} = \begin{pmatrix} 1-\lambda & 3 \\ 3 & 1-\lambda \end{pmatrix}.$$

Forming the determinant of the matrix above we obtain the following characteristic equations

$$(1-\lambda)^2 - 9 = \lambda^2 - 2\lambda + 1 - 9 = \lambda^2 - 2\lambda - 8 \equiv (\lambda + 2)(\lambda - 4),$$

which implies that we have two distinct eigenvalues: $\lambda_1 = -2$ and $\lambda = 4$. To find the associated eigenvectors for these eigenvalues, we have to form the matrix-vector equation $\mathbf{A}\mathbf{x}_i = \lambda_i \mathbf{x}$, for $i = 1, 2$. Rearranging this equation and factorizing the eigenvector, results in the following equation:

$$(\mathbf{A} - \lambda_i \mathbf{I})\,\mathbf{x_1} = \mathbf{0},$$

to solve. Therefore, substituting the first eigenvalue into the equation above yields

$$\left(\begin{pmatrix} 1 & 3 \\ 3 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}\right)\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} 3 & 3 \\ 3 & 3 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

$$\Rightarrow \mathbf{x}_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Following the same derivation above it can easily be shown that the second eigenvector, $\mathbf{x}_2$ for the matrix $\mathbf{A}$ is $\mathbf{x}_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. An important property of the eigenvalues is that they can be used to determine if a matrix is singular. We stated at the beginning of this section that the determinant is related to the product of the eigenvalues. Therefore, if we have a zero eigenvalue then the matrix is singular.

Exercise 2.14. *Find the eigenvalues and eigenvectors of the following matrices and calculate their determinants*

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 2 & 4 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 6 & 2 \\ 0 & 0 & 3 \end{pmatrix}.$$

Eigenvalues and eigenvectors play an important role in methods for transforming matrices. We consider these *decompositions* in the next subsection.

2.4 Matrix Decompositions

In this section we shall consider five different forms of decompositions of a matrix: Gaussian elimination and the LU decomposition, Cholesky, QR, diagonalization, and singular value decomposition. The first decomposition that we consider is LU decomposition which arises from what is referred to as **Gaussian Elimination**.

2.4.1 Gaussian Elimination and the LU Decomposition

Gaussian elimination is a technique for solving a set of linear simultaneous equations. We have alluded to some aspects of the techniques involved with Gaussian elimination when we explained about row and column reduction. If we have the matrix-vector equation $\mathbf{Ax} = \mathbf{b}$ to solve, then we wish to find a way to transform the matrix $\mathbf{A}$ into an upper triangular matrix, so that the new system of equations can be solved through the process of *back substitution*. As we just mentioned we have already described this technique for the row reduction of the matrix, but now we have to apply the same factorizations and subtractions to the vector $\mathbf{b}$.

The algorithmic description for the Gaussian elimination is as follow:

Step 1: We assume that the first diagonal entry of the starting matrix is not equal to zero. Given this assumption we define the row multipliers by

$$m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}}, \quad i = 2, 3, \ldots, n,$$

where the $a_{i1}^{(1)}$ are the current entries below $a_{11}^{(1)}$ which is the first diagonal entry.

Step 2: The second step is to eliminate the entries below the first leading diagonal entry in $\mathbf{A}$ but to apply the same subtraction to the entries in $\mathbf{b}$, which is expressed mathematically as

$$\begin{aligned} a_{ij}^{(2)} &= a_{ij}^{(1)} - m_{i1} a_{1j}^{(1)}, \quad i, j = 2, 3, \ldots, N, \\ b_i^{(2)} &= b_i^{(1)} - m_{i1} b_i^{(1)}, \quad i = 2, 3, \ldots, N. \end{aligned}$$

We repeat the two steps above for all $N - 1$ rows of the matrix $\mathbf{A}$, and the vector $\mathbf{b}$, below the first row until what remains is an upper triangular matrix-vector equation of the form

$$\begin{pmatrix} a_{11}^{(1)} & \cdots & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & \vdots \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & a_{NN}^{(N)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_N^{(N)} \end{pmatrix}.$$

The reason for introducing Gaussian elimination is that it plays a role in the formation of the LU decomposition. We have been able to reduce the matrix $\mathbf{A}$ into an upper triangular matrix $\mathbf{U}$ such that we are solving the matrix equation $\mathbf{Ux} = \mathbf{g}$, where the vector $\mathbf{g}$ is the collection of all of the altered entries of $\mathbf{b}$, that is to say $g_i = b_i^{(i)}$ for $i = 2, \ldots, N$.

We now have defined the **U** part of the LU decomposition, and so we move onto the L part which comes from the row multipliers m. We have a set of row multiplies for each round of the elimination and as such if we store these multipliers into a lower triangular matrix, then we have

$$\mathbf{L} \equiv \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ m_{21} & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ m_{N1} & m_{N2} & \cdots & \cdots & 1 \end{pmatrix}.$$

Now that we have both of the components of the LU decomposition we can now define the LU decomposition as.

Theorem 2.15. *If we have a lower triangular matrix* **L** *and an upper triangular matrix* **U** *that have been found through the Gaussian elimination method just described, then the matrix* **A** *can be expressed as*

$$\mathbf{A} = \mathbf{LU}.$$

The ability to write the full square matrix as the product of a lower and an upper triangular matrix is that it enables to describe some properties of the matrix **A**. The first of these properties is

$$\det(\mathbf{A}) = \det(\mathbf{L}) \det(\mathbf{U}),$$

which is equivalent to the product of the diagonal entries of the **U** matrix as the product of the diagonal entries of the **L** matrix is 1.

The next property relates to the inverse of **A**. If we are able to perform a Gaussian elimination such that $\mathbf{A} = \mathbf{LU}$, then the inverse of **A** is given by

$$\mathbf{A}^{-1} = (\mathbf{LU})^{-1} = \mathbf{U}^{-1}\mathbf{L}^{-1},$$

which plays an important part when trying to solve large matrix-vector equations which occur quite frequently in numerical modeling. However, to make the numerics of the LU decomposition more stable we may be required to perform **pivoting**. Pivoting is where we interchange rows or columns to have a better entry on the diagonal with which to perform the Gaussian elimination. This means that we have a permutation matrix **P**, that keeps a record of which rows and columns were interchanged. This then makes the decomposition of the form

$$\mathbf{PA} = \mathbf{LU},$$

which then makes the inverse of **A** as

$$(\mathbf{PA})^{-1} = (\mathbf{LU})^{-1} \Rightarrow \mathbf{A}^{-1} \equiv \left(\mathbf{P}^{-1}\mathbf{LU}\right)^{-1}.$$

2.4.2 Cholesky Decomposition

Before we summarize this useful decomposition, we need to introduce two important definitions: The first of these is the definition of a Hermitian matrix.

Definition 2.16. A matrix $\mathbf{A}$ that is in $\mathbb{C}^{N\times N}$ is said to be a Hermitian matrix if it is equal to its own conjugate transpose. This is implying that, the element in the $a_{i,j}$ entry is equal to the complex conjugate of the element in the a_{ji} entry. This must be true for all i and j.

An example of a Hermitian matrix is

$$\mathbf{A} = \begin{pmatrix} 1 & 2+i & 3 \\ 2-i & 4 & 3-2i \\ 3 & 3+2i & 2 \end{pmatrix}.$$

An important feature to note about Hermitian matrices is that their diagonal entries have to be real numbers as they must be their own complex conjugate. One way to interpret the Hermitian matrix is as the extension of symmetric matrices for real numbers to matrices with complex number entries, i.e., $\mathbf{A}^H = \mathbf{A}$.

The second definition we require is that of definiteness of a matrix.

Definition 2.17. There are four possible versions of a different form of definiteness for real, symmetric matrices, and by association for the complex Hermitian matrices as follows:

1. A symmetric $N \times N$ real matrix $\mathbf{A}$ is said to be **positive definite** if the scalar that arises from $\mathbf{z}^T\mathbf{A}\mathbf{z}$ is positive for every non-zero column vector $\mathbf{z} \in \mathbb{R}^N$. That is to say $\mathbf{z}^T\mathbf{A}\mathbf{z} > 0$.
2. A symmetric $N \times N$ real matrix $\mathbf{A}$ is said to be **positive semidefinite** if the scalar that arises from $\mathbf{z}^T\mathbf{A}\mathbf{z}$ is always greater than or equal to zero $\forall \mathbf{z} \in \mathbb{R}^N$. That is to say $\mathbf{z}^T\mathbf{A}\mathbf{z} \geq 0$.
3. A symmetric $N \times N$ real matrix $\mathbf{A}$ is said to be **negative definite** if the scalar that arises from $\mathbf{z}^T\mathbf{A}\mathbf{z}$ is negative for every non-zero column vector $\mathbf{z} \in \mathbb{R}^N$. That is to say $\mathbf{z}^T\mathbf{A}\mathbf{z} < 0$.
4. A symmetric $N \times N$ real matrix $\mathbf{A}$ is said to be **negative semidefinite** if the scalar that arises from $\mathbf{z}^T\mathbf{A}\mathbf{z}$ is always less than or equal to zero $\forall \mathbf{z} \in \mathbb{R}^N$. That is to say $\mathbf{z}^T\mathbf{A}\mathbf{z} \leq 0$.

The general definition for the different forms of definiteness, which includes the Hermitian matrices, is that a $N \times N$ Hermitian matrix is said to be positive definite if

$$\mathbf{z}^H\mathbf{A}\mathbf{z} > 0, \quad \forall \mathbf{z} \in \mathbb{C}^N. \tag{2.44}$$

The inequality in (2.44) can easily be changed to obtain the complex matrix definition of positive semidefiniteness, negative definiteness and negative semidefiniteness.

The reason for introducing these definitions above is because the Cholesky decomposition can only be applied to Hermitian semidefinite matrices.

Definition 2.18. For a Hermitian semidefinite matrix $\mathbf{A}$, the Cholesky decomposition of $\mathbf{A}$ is defined as

$$\mathbf{A} \equiv \mathbf{L}\mathbf{L}^H, \tag{2.45}$$

where $\mathbf{L}$ is a lower triangular matrix and $\mathbf{L}^H$ is the conjugate transpose of $\mathbf{L}$.

Some important properties of the Cholesky decomposition are as follows:

- Every Hermitian positive-definite matrix has a unique Cholesky decomposition.
- If the matrix $\mathbf{A}$ is Hermitian but is positive semidefinite, then it is still true that $\mathbf{A} = \mathbf{L}\mathbf{L}^H$ if the diagonal entries of $\mathbf{L}$ are equal to zero.
- If the matrix $\mathbf{A}$ is a real symmetric positive definite matrix then $\mathbf{L}$ is a real numbers lower triangular matrix and the Cholesky decomposition becomes $\mathbf{A} \equiv \mathbf{L}\mathbf{L}^T$.

- For positive definite matrices, the Cholesky decomposition is unique. That is to say that there exists only one lower triangular matrix **L** with strictly positive entries on the diagonal that satisfies the definition above.
- However, the same is not true if the matrix is positive semidefinite. There still exists a Cholesky decomposition; however, there exists more than one lower triangular matrix that satisfies the decomposition.

The Cholesky decomposition is a more efficient version of Gaussian elimination, and is known to be twice as efficient as the LU decomposition if you are able to know that the matrix in your matrix-vector system of linear simultaneous equations is Hermitian positive definite. The Cholesky decomposition is used in many forms of minimization, least squares fits, some forms of Kalman filtering, as well as in some forms of Monte-Carlo modeling, all of which we shall go into more details in later chapters.

2.4.3 The QR Decomposition

The QR decomposition of a matrix **A** is defined as follows:

Definition 2.19. The QR decomposition of the rectangular $M \times N$ matrix, **A**, is given by

$$\mathbf{A} \equiv \mathbf{QR}, \tag{2.46}$$

where the matrix **Q** is an orthogonal $M \times M$ matrix, such that $\mathbf{Q} \in \mathbb{R}^{M \times M}$, and $\mathbf{R} \in \mathbb{R}^{M \times N}$ is a real numbered upper triangular matrix.

If matrix **A** has full column rank, then the first N columns of **Q** form an orthonormal basis for the range of **A**. Therefore, the calculation of the QR decomposition is a way to compute an orthonormal basis for a set of vectors. There are several different methods to calculate the QR decomposition; we recommend [157] for a more detail description of these methods.

2.4.4 Diagonalization

We return to eigenvalues and eigenvectors to define a decomposition involving them. This decomposition is called diagonalization as it refers to the process that results in a diagonal matrix. Before we define diagonalization, let us introduce the following definition of similarity between matrices.

Definition 2.20. Let **A** and **B** be two square matrices that are of the same dimensions; the matrix **A** is said to be **similar** to the matrix **B** if there is a non-singular matrix **P**, such that

$$\mathbf{B} \equiv \mathbf{P}^{-1}\mathbf{AP}.$$

Some useful, and important, properties of similar matrices are:

- If the matrices **A** and **B** are similar, then the characteristic equations for their eigenvalues are the same.
- The eigenvalues of similar matrices are the same.
- The traces of **A** and **B** are the same, as well as their determinants.

As the title of this subsection suggests we are looking for a method to transform the matrix **A** into a diagonal matrix **D**. This similarity transform, or **canonical form**, is referred to as **diagonalization**. The matrix that this transform can be applied to is said to be a **diagonalizable matrix**.

Definition 2.21. A square matrix $\mathbf{A}$ is said to be diagonalizable if there exists a matrix $\mathbf{P}$ such that

$$\mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \mathbf{D}. \tag{2.47}$$

If the matrix $\mathbf{A}$ has a full set of eigenvectors then the matrix $\mathbf{P}$ is the matrix containing these eigenvectors and the diagonal matrix $\mathbf{D}$ is equivalent to $\mathbf{\Lambda}$ which is a diagonal matrix with the eigenvalues of $\mathbf{A}$ as the diagonal entries.

Example 2.22. *For the matrix* $\mathbf{A} = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 3 & 0 \\ 2 & -4 & 2 \end{pmatrix}$*, find the eigenvalues and eigenvectors and show that* $\mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \mathbf{\Lambda}$.

We know that to calculate the eigenvalues we need to find the characteristic polynomial of $\mathbf{A}$ which yields $\begin{pmatrix} 1-\lambda & 2 & 0 \\ 0 & 3-\lambda & 0 \\ 2 & -4 & 2-\lambda \end{pmatrix}$. The first thing to notice is that we can form the determinant through expanding about the third column, which yields an upper triangular submatrix who's determinant is the product of the diagonal entries therefore, the determinant of $\mathbf{A} - \lambda\mathbf{I}$ is $(1-\lambda)(3-\lambda)(2-\lambda)$, which implies that the three distinct eigenvalues are $\lambda_1 = 1$, $\lambda_2 = 2$, and $\lambda_3 = 3$. The corresponding eigenvectors can easily be shown to be $\mathbf{v}_1 = \begin{pmatrix} -1 \\ 0 \\ 2 \end{pmatrix}$, $\mathbf{v}_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, and $\mathbf{v}_3 = \begin{pmatrix} -1 \\ -1 \\ 2 \end{pmatrix}$.

An important feature to note here is that it does not matter which order you place the eigenvectors into the matrix $\mathbf{P}$ their corresponding eigenvalue will appear on that associated diagonal entry of $\mathbf{\Lambda}$. Therefore, if we wish the eigenvalues to appear in increasing order we would form $\mathbf{P} = \begin{pmatrix} -1 & 0 & -1 \\ 0 & 0 & -1 \\ 2 & 1 & 2 \end{pmatrix}$; if we would like the eigenvalues to be in descending order, then we would form $\mathbf{P} = \begin{pmatrix} -1 & 0 & -1 \\ -1 & 0 & 0 \\ 2 & 1 & 2 \end{pmatrix}$.

Taking the $\mathbf{P}$ matrix for the ascending eigenvalue situation, we now require the inverse of $\mathbf{P}$. The first step is to find the determinant, which upon expanding the second row yields that $|\mathbf{P}| = -1$. Given this, and recalling the order of the sign for the cofactors it can easily be shown that $\mathbf{P}^{-1} = \begin{pmatrix} 1 & -1 & 0 \\ 2 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}$.

Exercise 2.23. *Given the definitions for* $\mathbf{P}$ *and* $\mathbf{P}^{-1}$ *above, verify that* $\mathbf{P}^{-1}\mathbf{A}\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$.

2.4.5 Singular Value Decomposition

Another important decomposition of matrices is referred to as the **singular value decomposition**, often referred to as SVD, and is defined as

Definition 2.24. Given a real $M \times N$ matrix $\mathbf{A}$, then there exist orthogonal matrices

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_M \end{bmatrix}, \text{ and } \mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_N \end{bmatrix},$$

where $\mathbf{U} \in \mathbb{R}^{M \times M}$ and $\mathbf{V} \in \mathbb{R}^{N \times N}$, such that

$$\mathbf{U}^T \mathbf{A} \mathbf{V} = \text{diag}\left(\sigma_1, \sigma_2, \ldots, \sigma_k\right),$$

where $k = \min\{M, N\}$ and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0$. The σ_i, for $i = 1, 2, \ldots, k$ are the singular values, while $\mathbf{u}_i$ is the ith left singular vector and $\mathbf{v}_i$ is the ith right singular vector.

Because the $\mathbf{U}$ and $\mathbf{V}$ matrices are orthonormal it is possible to rearrange the decomposition such that $\mathbf{A}$ can be expressed as

$$\mathbf{A} \equiv \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^T,$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix that contains the associated singular values on the diagonal.

An important feature to note about the singular value decomposition is that it can be applied to rectangular matrices, whereas the eigenvalue decomposition (the diagonalization presented in the last subsection) can only be applied to square matrices. However, it is possible to relate the two different decomposition through

$$\mathbf{A}^H \mathbf{A} = \mathbf{V} \boldsymbol{\Sigma}^H \mathbf{U}^H \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^H = \mathbf{V} \left(\boldsymbol{\Sigma}^H \boldsymbol{\Sigma} \right) \mathbf{V}^H,$$
$$\mathbf{A} \mathbf{A}^H = \mathbf{U} \boldsymbol{\Sigma} \mathbf{V}^H \mathbf{V} \boldsymbol{\Sigma}^H \mathbf{U}^H = \mathbf{U} \left(\boldsymbol{\Sigma} \boldsymbol{\Sigma}^H \right) \mathbf{U}^H,$$

where the columns of $\mathbf{V}$, the right singular vectors, are the eigenvectors of $\mathbf{A}^H \mathbf{A}$, and the columns of $\mathbf{U}$, the left singular vectors, are the eigenvectors of $\mathbf{A} \mathbf{A}^H$.

2.5 Sherman-Morrison-Woodbury Formula

The Sherman-Morrison-Woodbury formula is often used in the derivations of different solutions to the different forms of data assimilation. The formula is stated as follows: given the matrices $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{U}$ and $\mathbf{V} \in \mathbb{R}^{n \times k}$, then the inverse of $\left(\mathbf{A} + \mathbf{U}\mathbf{V}^T\right)$ is

$$\left(\mathbf{A} + \mathbf{U}\mathbf{V}^T\right)^{-1} \equiv \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\left(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^T\mathbf{A}^{-1}, \tag{2.48}$$

where we have assumed that $\mathbf{A}$ and $\left(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}\right)$ are both invertible. There are many different manipulations of (2.48) that are used in deriving different formulations of data assimilation analysis equations. The first formulation involves the square root of the matrix $\mathbf{A}$ denoted as $\mathbf{A}^{\frac{1}{2}}$, where you pre multiply by the square root matrix and post multiply by its transpose. Another version is obtained by pre-multiplying by $\mathbf{A}$. Given the two manipulations just described, the resulting versions of (2.48) are

$$\mathbf{A}^{1/2}\left(\mathbf{A} + \mathbf{U}\mathbf{V}^T\right)^{-1}\mathbf{A}^{T/2} \equiv \mathbf{I} - \mathbf{A}^{-1/2}\mathbf{U}\left(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^T\mathbf{A}^{-T/2}, \tag{2.49}$$

$$\mathbf{A}\left(\mathbf{A} + \mathbf{U}\mathbf{V}^T\right)^{-1} \equiv \mathbf{I} - \mathbf{U}\left(\mathbf{I} + \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^T\mathbf{A}^{-1}, \tag{2.50}$$

respectively. Note that both **U** and **V** could be the product of several matrices that result in rectangular matrices.

2.6 Summary

In this chapter we have provided a brief introduction to some aspects of linear algebra that play an important part in different parts of numerical modeling, minimization and data assimilation to name but a few areas. We have defined several different forms of matrix decompositions that will be useful as we develop the different tools that make up many of the different components of data assimilation.

In the following chapter, we move on to the very important topic of univariate distribution theory, which will lay the foundations for the probabilistic framework of most data assimilation schemes.

This page intentionally left blank

CHAPTER

Univariate Distribution Theory

3

Contents

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00007-3

Univariate statistical analysis is based on there being one random variable. However, even from this arguably simple point of view, a lot of insights can be learned on how to extend to multiple random variables. This section introduces the theory, and gives examples, of discrete random variables, continuous random variables, as well as the three important descriptive statistics, mean, mode, and median, along with the higher moments of variance, skewness, and kurtosis. We shall also introduce many different probability density and mass functions that are used for modeling of geophysical processes. We start with introducing the definition of a random variable.

3.1 Random Variables

The starting point to define discrete random variable theory is from what are called **variates**. A definition for a variate in [64] is as follows:

Definition 3.1. A **variate** is any quantity or attribute whose value varies from one unit of investigation to another.

Variates can be identified further as quantitative or qualitative; again from [64] as

Definition 3.2. A **quantitative variate** is a variate whose values are numerical.

Definition 3.3. A **qualitative variate** is a variate whose values are not numerical.

It is within quantitative variates that we find the split between continuous and discrete variates, which are defined as follows.

Definition 3.4. A **continuous** variate is a variate that may take all values within a range.

Definition 3.5. A **discrete** variate is a variate whose values change by steps.

Examples of these different types of variates from an atmospheric science perspective could be for the discrete quantitative variate. The number of a certain category of hurricane making landfall is an example of a variable that can only be an integer, we cannot have half a hurricane making landfall—or at least it is not counted as such. An example of a meteorological continuous quantitative variate would be the maximum wind speed at landfall of the hurricane or tropical storm/depression. The magnitude of the wind could be any value to any decimal point, even if we cannot observe the wind speed to that accuracy. Another example associated with tropical cyclones would be the barometric pressure of the tropical cyclone at landfall.

Examples in other geosciences of the two different types of quantitative variates would be size of algae bloom compared to the number of blooms per month or in a specific geographical location for oceanography. A cryosphere example of a discrete variate is number of days of snow cover in a location, while a continuous variate is the amount of snow water equivalent or snow depth. For hydrology, an example of a discrete variate is the total number of rainstorms in a specific capture basin, while a continuous variate is the amount of rain that fell in each storm, the amount of snow melt into a river system or the amount of water passing a specific point in the river over a certain time period. Examples in seismology are the magnitude of a earthquake, which would be a continuous variate, and for the discrete variate case, the number of earthquakes in a specific month or a geographical location.

Table 3.1 Hypothetical Frequency Distribution of the Category of Landfalling Tropical Cyclones for the Atlantic Seaboard and the Gulf of Mexico Between 1984 and 2014.

	TD	TS	CAT 1	CAT 2	CAT 3	CAT 4	CAT 5	TOTAL
Frequency	3	5	10	15	25	17	12	87
Relative frequency	0.0345	0.0575	0.1149	0.1724	0.2874	0.1954	0.1379	1.0000

If we take the atmospheric science example of the number of each category of hurricane, and of each tropical storm or depression if we cover the whole range of tropical cyclones, we could then identify qualitative discrete variates for states along the Gulf of Mexico coast. As Texas has a longer coastline than the other states of Louisiana, Mississippi, Alabama, and Florida, this could mean that there is a higher probability that Texas will experience cyclones. You may wish to subset that total number of landfalling tropical cyclones in a specific state to the category of those landfalling tropical cyclones. A fairer qualitative variate could be month of the year.

Given the variates that you have decided to observe or to analyze, the next important statistical definition is for the **frequency distribution**:

Definition 3.6. The **frequency distribution** of a discrete variate is the set of possible values of the variate, together with the associated frequencies.

Returning to the landfalling hurricanes, if we now consider the total number of landfalling tropical cyclones on the Atlantic seaboard combined with the Gulf of Mexico between 1984 and 2014, we could have the following frequency distribution. **Note:** This is not the actual data for the total number of landfalling tropical cyclones during this 30-year period. We have presented this synthetic frequency distribution in Table 3.1.

3.2 Discrete Probability Theory

We begin the derivation of discrete probability theory with the definition of a **population**.

Definition 3.7. A **population** is a collection of items that are under discussion. A population can be finite or infinite; it can be an actual physical feature or it can be a hypothetical feature.

A sample of a population can be considered as any subset of the population. There are different types of samples that can be taken of the population but the most simplest, and most often used, is the **random sample**. The definition of a random sample is as follows.

Definition 3.8. A **random sample** is a sample of the population chosen so that every member of the population is equally likely to be a member of the sample, independently of which other members of the population are chosen.

Given the definitions of sampling and population, we can start to consider the motivation for probability theory for discrete variates. Probability theory was initially developed to help with gambling: the chances of certain cards being drawn from a deck or the chances of a certain number on a die or combinations over multiple dice and the simplest of the all, a coin toss. In essence, the probability of an event is the relative frequency of that event occurring.

As an example, suppose we are considering randomly selecting for analysis a hurricane test case from our sample of just hurricanes over the 30-year period 1984–2014. What is the probability that we select a major hurricane to analyze? We know from Table 3.1 that there were 400 storms in total in

the 30 years studied; we also know that there were 80 major hurricanes in that period and 320 smaller hurricanes. The relative frequencies are $\frac{80}{400}$ and $\frac{320}{400}$ for major hurricanes and smaller hurricanes, respectively. If we take one sample of one hurricane, then the probability of the hurricane picked being a major hurricane is equivalent to the relative frequency.

If, however, we returned the picked hurricane back into the population of hurricanes and then drew another single sample, recorded the type of hurricane, and then returned again to the population, we would hope that as the number of times we sample the population of hurricane is large enough, the proportion of the samples that are major hurricanes should approach the relative frequency given above. This form of sampling is known as **sampling with replacement**.

The basic definition of the *probability* of selecting a major hurricane is the ratio of the number of ways of selecting a major hurricane (80) to the number of ways of selecting a hurricane (400). There are bounds of what value a probability can have. If all 400 hurricanes in the example above had been major hurricanes, then the probability of selecting a major hurricane would be $\frac{400}{400} = 1$. On the other hand, if all the hurricanes in the 30-year period had been weaker ones, then the probability of selecting a major hurricane would be $\frac{0}{400} = 0$. While this may appear trivial, it is the starting point of probability theory, and it gives us our first properties for probability as follows.

Definition 3.9. A probability, p, can only take a value between 0 and 1, i.e., $p \in [0, 1]$.

If we set p to be the probability of selecting a major hurricane, then the probability of not selecting a major hurricane is $1 - p$.

The next step toward probability theory is to consider a **trial** which is loosely defined as a process that, when repeated, generates a set of observation results. The next definition, which again may appear obvious, is that the **outcome** is the result of the trial. Finally an **event** is a set that contains one or more outcomes of a trial. Therefore, an event is a subset of the possible outcomes.

A classic mathematical example to illustrate the terms above is to consider a fair coin (where either side of the coin is equally likely to occur) tossed three times. The gambler wins if a head is shown. As shown in Fig. 3.1, where a tree diagram illustrates the likely events, we can see that there are eight possible outcomes at the end of the three tosses. Therefore, we know that the denominator of the equivalent ratio to the one defined above for the hurricane example is 8. An example of an event for the coin toss would be that the gambler won two of the coin tosses. If we represent the outcomes

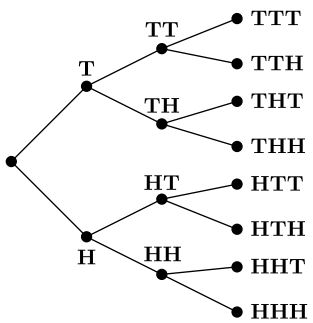


FIGURE 3.1

Example of a tree diagram for a fair coin toss.

of the coin tosses by H for head and T for tail, then the event space would be $\{HHT, HTH, THH\}$. Another event could be that the gambler wins the middle toss, which would have the event space: $\{HHT, THT, THH, HHH\}$.

Often it is quite illustrative to display an analysis that we are considering in the form of a diagram. For probability theory, it is useful to consider events in the form of sets and then to display these sets in the form of a diagram. One diagram that displays different sets of outcomes is known as a *Venn diagram*. The Venn diagram consists of a set of points which represent all of the equally likely events, and then the different events are represented as subsets of the points. Often the set of all the possible outcomes is called the universal or universe set. This complete set can also be referred to as the **possibility space** of the **sample space**. The complete space of outcomes will be denoted as S.

We label the two events mentioned above as A: the gambler wins two of the three coin tosses and B: the gambler wins the middle toss, and finally introduce the third event, C: the gambler losses all three tosses. The event space for C is $\{TTT\}$.

In Fig. 3.2, we have drawn the possibility space for the three coin toss outcomes. As we can see from Fig. 3.2, event C only has one entry, while events A and B have multiple entries. The points in the possibility space S are referred to as elementary events. As we can see, event C is also an elementary event as it only contains the one outcome TTT. The other two events are compound events, which then include several separate events.

We now introduce the notation N to represent the number of events. The expression $N(S)$ represents the total number of events in the possibility space, while $N(E)$ represents the number of possible outcomes in the event E. Given these two expressions, it is now possible to give a more mathematical definition of the probability.

Definition 3.10. If a trial has a set of outcomes that are equally likely, then the probability of the event E, $p(E)$, is given by

$$p(E) \equiv \frac{N(E)}{N(S)}.$$

The next step after defining the probability is to consider how likely the event is. In the coin toss example, the gambler is interested in the probability of winning 0, 1, 2, 3 tosses. This means that there are four possible outcomes at the end of the trial. The total number of outcomes that can lead to each of the events just mentioned are 1, 3, 3, and 1, respectively. Therefore, Definition 3.10 implies that the probability for each outcome is $\frac{1}{8}$, $\frac{3}{8}$, $\frac{3}{8}$, $\frac{1}{8}$. The probabilities assigned to the four events of the number of coin tosses won each represent a *mass* associated with each event and are called **probability**

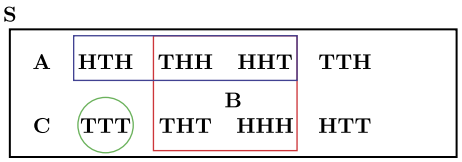


FIGURE 3.2

Possibility space for events A: Gambler wins two tosses, B: Gambler wins the middle toss, and C: Gambler wins no tosses.

masses. Therefore, the probability mass of an event E, denoted as $M(E)$, is equivalent to the sum of the probability masses assigned to each point in event E. If we consider the case of the events that the gambler wins one or two tosses, then this is equivalent to the probability of 1 or 2, which is $\frac{1}{8} + \frac{3}{8} = \frac{1}{2}$.

The procedure explained above to determine probabilities of events and the possibility space, can be summarized as follows and is referred to as the **axioms of probability**. This will eventually lead to the theory that enables us to define data assimilation in the variational, ensemble, Monte Carlo, and particle formulations.

Definition 3.11

1. Specify the points in the possibility space.
2. Assign a probability to each point in the possibility space; note that it is quite often the case that each point is not equally likely.
3. Calculate the probability of the event E by summing the probability masses assigned to the points in the event E.

The example and the theory developed so far in this subsection are for the discrete case, where there are a finite number of outcomes, and therefore the possibility space is referred to as the discrete possibility space. A formal definition for the discrete possibility space is as follows.

Definition 3.12. A discrete possibility space, denoted S, consists of n elementary events denoted by $s_1, s_2, \ldots, s_n$.

Given Definition 3.12, it is possible to define the probability function:

Definition 3.13. A probability function is defined on the possibility space S such that the following conditions hold:

1. $P(s_i) > 0, i = 1, 2, \ldots, n$.
2. $\sum_{i=1}^{n} P(s_i) = 1$.
3. For any of the events E in S then $P(E) = \sum_{s_i \in E} P(s_i)$.

However, it may become important to consider when events intersect one and another, and the associated probability of that intersection. It is assumed that by the term "intersection," we refer to the number of points that are in both or multiple intersecting events. If we consider the Venn diagram in Fig. 3.2, then we can see that the number of points in the intersection of events A and B are denoted as $N(A \cap B) = 2$. The two points that are in the intersection of events A and B are HHT and THH. To find the probability of the intersection of events A and B, we would consider the basic formula from earlier which would give $P(A \cap B) = \frac{2}{8} = \frac{1}{4}$.

If we now consider the intersection of events B and C, then we see that no points are contained in both events. This implies that $N(B \cap C) = 0$, or, if we were considering set theory, then $B \cap C = \emptyset$, where $\emptyset$ is referred to as empty of null set. When there are no points in the intersection of two events, they are said to be **mutually exclusive**.

The next probability operator that we consider after the intersections is the union of events. The points in the union of two events are the points that are in event A or event B, and those in their intersection (if there is one). The notation for the union of two events is $A \cup B$. Returning to the three events above, the union of events A and B would be $A \cup B = \{HHH, HHT, HTH, THH, THT\}$, which in words would be: what is the probability that the gambler wins either two tosses **or** that they win the middle toss? For the intersection, we were seeking the probability that the gambler won two tosses **and** that one of those wins was the middle toss.

We now consider a couple of theorems that are important for probability theory and for the derivation of the data assimilation formulations.

Theorem 3.14. *If we have two mutually exclusive events, U and V, of a discrete space S, then the probability of the union of U and V, $P(U \cup V)$, is*

$$P(U \cup V) = P(U) + P(V). \tag{3.1}$$

Theorem 3.15. *If we have two not mutually exclusive events, U and V, of a discrete space S, then the probability of the union of U and V, $P(U \cup V)$, is*

$$P(U \cup V) = P(U) + P(V) - P(U \cap V), \tag{3.2}$$

where the subtraction of the intersection is to remove the double counting of the points that are in both events.

All the theory and examples above can be extended to more than two events. If we have n events, E_i, for $i = 1, 2, \ldots, n$, in the discrete possibility space, and if they are all mutually exclusive, which implies that no events share any points, then the probability of the union of the n events is

$$P(E_1 \cup E_2 \cup \cdots \cup E_K) = P(E_1) + P(E_2) + \ldots P(E_n) \equiv \sum_{i=1}^{n} P(E_i).$$

If, however, the events are not mutually exclusive, then the intersections of events will have to be subtracted. For example, if we had four events, A, B, C, and D, and we know that A is mutually exclusive to the other three events, and we know that D is not mutually exclusive of C but is of B, and we know that B is not mutually exclusive of C, then $P(A \cup B \cup C \cup D)$ would be

$$P(A \cup B \cup C \cup D) = P(A) + P(B) + P(C) + P(D) - P(C \cap D) - P(B \cap C).$$

We now arrive at the first important tool of probability theory that is directly related to how data assimilation is derived; **conditional probability**.

Conditional probability is where we seek the probability of two or more events occurring, but before we determine the second event, we are aware that the first event has occurred.

If we have the two events U and V, then the **conditional probability** of V given U is the probability that event V will occur given that U has occurred. Conditional probabilities are denoted as $P(V|U)$.

To derive the mathematical expression that enables us to quantify the conditional probability, we first consider the part of the possibility space where the event U occurs. Therefore we now have that the new possibility set are the points in the event U. Given this new possibility space, all of the points inside this space are scaled by $P(U)$. The next step is to seek the points in the intersection of V and U. Therefore, the definition of conditional probability is given by

$$P(V|U) = \frac{P(U \cap V)}{P(U)}. \tag{3.3}$$

An important rearrangement of (3.3) is to obtain an expression for the intersection of two events as the product of the conditional probability and what is referred to as the **marginal probability**, denoted as,

$$P(U \cap V) = P(U)\, P(V|U). \tag{3.4}$$

The next step in probability theory is to consider a situation where events are **independent**. If we have two events, and they provide no information about each other, and if we consider the expression in (3.4), then we have $P(V|U) = P(V)$. This leads to the important property for the intersection of the events U and V as $P(U \cap V) = P(U)\,P(V)$. Thus, we can now define independent events as:

Definition 3.16. The events U and V are independent if and only if

$$P(U \cap V) = P(U)\,P(V).$$

It can easily be shown that Definition 3.16 can be extended to three or more events. If we have a third event, W, and we know that U, V, and W are independent, then

$$P(U \cap V \cap W) \equiv P(U \cap (V \cap W)) = P(U)\,P(V \cap W) = P(U)\,P(V)\,P(W).$$

This then leads to the following theorem.

Theorem 3.17. *If n events, $E_1, E_2, \ldots, E_n$ are independent of each other, then the probability of the intersection of all n events is*

$$P(E_1 \cap E_2 \cap \cdots \cap E_n) = P(E_1)\,P(E_2)\ldots P(E_n) \equiv \prod_{i=1}^{n} P(E_i). \tag{3.5}$$

Given all the definitions above for conditional probabilities and intersections, it is now possible to derive the fundamental building block for data assimilation: **Bayes' theorem**.

We start by considering two events: A and B. We know that the conditional probability of A given B is

$$P(A|B) = \frac{P(B \cap A)}{P(B)}.$$

Next note that the intersection operator is commutative, which means $P(B \cap A) = P(A \cap B)$. However, that now means that the conditional probability of B given A can be written as

$$P(B|A) = \frac{P(A \cap B)}{P(A)}.$$

Multiplying throughout by $P(A)$ to obtain an expression for $P(A \cap B)$ yields

$$P(A \cap B) \equiv P(B|A)\,P(A).$$

Substituting the expression above into the first conditional probability expression above results in

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}, \tag{3.6}$$

which is the two-event version of Bayes' theorem.

3.2.1 Discrete Random Variables

So far we have considered a situation where we have actual data to determine a variate's values and the probability of that variate occurring. However, we may often wish to predict a value for the variate. There are two types of models that can be used to obtain this goal: a physical model, where we substitute something else to represent the variate, or a theoretical model. It is of course the latter that we use in data assimilation and we shall follow this path now.

A theoretical model is based on using some form of observed relative frequency for the variate as the basis for the probability of the variate occurring. By labeling the variate in the hypothetical model with attributes, this variate is considered a **random variable**. This gives us the rough definition for a random variable as follows.

Definition 3.18. A random variable is any quantity whose values vary from one unit of a hypothetical population to another.

The definition above is similar to that of a discrete variate, but instead of investigating a population, we now consider a hypothetical population. The next important link from the discrete variates to random variables is the extension of the frequency distribution. The equivalent of the frequency distribution for a random variable is referred to as the **probability distribution**, and this forms the basis of the data assimilation algorithms presented later.

An example of a basic probability model is the **binomial** probability model. If we have a discrete random variable, z, that has two attributes, a and b, that have an associated frequency distribution of 0.2 and 0.8, respectively, then it is possible that you wish to make a prediction about z taking either value a or b, say out of a sample of 20. The binomial probability model is based on the probability of success and failure. If we associate success with $z = a$ and failure with $z = b$, then we have to consider all of the possible combinations out the sample of 20 that each attribute could occur, assuming that each event is mutually exclusive. We now introduce the notion of permutations, which will enable us to assign probabilities to the different possible outcome of the sample of 20.

Permutations are ways of arranging n objects of which r are of one type and $(n - r)$ are of the other type. The method of ascertaining how many ways there are of selecting different values for r out of a sample of size n is

$$\frac{n!}{r!\,(n-r)!} = \left(\begin{array}{c} n \\ r \end{array} \right), \tag{3.7}$$

where the ! operator is called the factorial operator, and is defined as

$$n! \equiv \prod_{i=1}^{n} i. \tag{3.8}$$

Note: The value of 0! is 1.

If we have a situation where $n = 5$ and $r = 2$, then the total number of ways of selecting two successes out of a sample is

$$\left(\begin{array}{c} 5 \\ 2 \end{array} \right) = \frac{5!}{2!3!} = \frac{5 \times 4 \times 3 \times 2 \times 1}{2 \times 13 \times 2 \times 1} = \frac{120}{2 \times 6} = 10.$$

We return to the theoretical example of the discrete random variable z. The permutation definition enables us to derive the total combinations from our sample of 20 where we can have $z = a$ or $z = b$.

Given the associated probability for $z = a$ and $z = b$ of 0.2 and 0.8, respectively, the binomial probability model for this situation is:

$$P(r) = \binom{n}{r} (0.2)^r (0.8)^{n-r}, \quad r = 0, 1, 2, \ldots, 20.$$

Therefore, the final probability distributions for the random variable z are shown in Table 3.2.

The two different methods for describing the possible values of the discrete random variable, r, and the associated probability, in either the form of a table, or a mathematical expression, are both referred to as ***Probability Mass Function*** (PMF).

An important feature of the PMF is that it enables us to define the ***cumulative distribution function*** (CDF). The CDF is used when we wish to ascertain the probability of the discrete random variable being less than or equal to a specific value. As we have assumed that each value is independent, we are therefore seeking the union of all of the probabilities of the values for r up to and including b. The algebraic way of expressing this problem is $P(r \leq b)$. If we return to our simple problem above, and consider that we would like to see the probability of $r \leq 4$, then we know that the probability mass function is defined by $f(r) = \binom{n}{r} (0.2)^r (0.8)^{n-r}$, but that we need the sum from $r = 0$ to $r = 4$. Therefore, we require

$$\begin{aligned} P(r \leq 4) = &\binom{20}{0} (0.2)^0 (0.8)^{20} + \binom{20}{1} (0.2)^1 (0.8)^{19} + \binom{20}{2} (0.2)^2 (0.8)^{18} \\ &+ \binom{20}{3} (0.2)^3 (0.8)^{17} + \binom{20}{4} (0.2)^4 (0.8)^{16}. \end{aligned} \tag{3.9}$$

We can see that we can write the expression in (3.9) in a generalized summation form as

$$P(r \leq 4) = \sum_{i=1}^{4} \binom{20}{i} (0.2)^i (0.8)^{20-i}. \tag{3.10}$$

The expression in (3.10) can be generalized to any sample size, and probability. Now we shall denote the probability of a success as τ, while the probability of failure is denoted as $1 - \tau$, which is referred to as the **compliment** probability. Therefore, we can write the PMF and the CDF for the simple binomial probability model as

$$f(n, r) \equiv \binom{n}{r} (\tau)^r (1 - \tau)^{n-r}, \quad r = 0, 1, 2, \ldots, n, \tag{3.11}$$

$$F(r \leq b) \equiv \sum_{i=0}^{b} \binom{n}{i} (\tau)^i (1 - \tau)^{n-i}. \tag{3.12}$$

The PMF in (3.11) in generalized form is called the **binomial distribution**.

Table 3.2 Binomial Probability Model Example.

$z = a$	0	1	2	3	4	5	6	7	8	≥ 9
Probability	0.01	0.06	0.14	0.21	0.22	0.17	0.11	0.06	0.02	0.00

3.3 Continuous Probability Theory

Earlier in this chapter we mentioned continuous variate, and so now we shall expand on what the difference is between a continuous and discrete variate, and follow through the probability derivation for the continuous random variable to obtain the associated probability distribution and cumulative distributions functions.

First we recall that a discrete variate can take a finite set of values, whereas a continuous variate is such that it can take any value within a given range. To build the structure for continuous random variables, we have to define the equivalent of the frequency distribution from the discrete variate. We start with two definitions, which are as follows.

Definition 3.19. A **class interval** is a subdivision of the total range of values that a continuous variate could take.

Definition 3.20. A **class frequency** is the number of observations of the continuous variate in a given interval.

Given these two definitions, the continuous version of the frequency distribution for a continuous variate is defined as:

Definition 3.21. A **frequency distribution** of a continuous variate is the set of class intervals for the variate, combined with the associated class frequency.

A synthetic meteorological example here is that of wind speed that could be observed at a weather station, for a whole day. The winds speeds are *recorded* every five minutes, but the observed wind speeds can take any value between 0 mph and maybe 100 mph. For this data set we have to consider what class intervals we wish to select. We have 144 observations of the wind speed ranging from 0 mph to approximately 17.8 mph, so a suggestion for the class intervals would be 1 mph. Therefore, we have 18 intervals where any wind speed between 0 and 0.9 of the integer is recorded in that interval.

A common way to understand the behavior of a variate is through a graphical representation. This is also an important approach for verify some of the assumptions about the expected performance of data assimilation systems. The most frequently used visualization is a histogram of the data. Fig. 3.3 contains a histogram of the wind observations from Table 3.3. We can see that there appear to be a lot more instances of wind speeds under 3 mph. However, there were observations of wind speeds of 0 mph. This is an important feature as this indicates that wind speed is a **positive semi-definite** random variable, which means that the variate appears to be able only to take a value greater than or equal to zero.

As with the discrete variates, we need to develop a model to possible predict future values, given the probability associated with that value. However, the problem with a continuous variate is that we have binned the observations into intervals, as there may only be one or two actual observations of a specific value, plus the wind speed could take a value to an infinite decimal place, but we cannot measure to

Table 3.3 Synthetic Wind Speeds.

Interval (mph)	0–0.99	1–1.99	2–2.99	3–3.99	4–4.99	5–5.99
Frequency	31	33	34	14	10	10
Relative frequency	0.215	0.229	0.236	0.097	0.069	0.069
Interval (mph)	6–6.99	7–7.99	8–8.99	9–9.99	10–10.99	11–11.99
Frequency	6	2	1	2	0	1
Relative frequency	0.042	0.014	0.007	0.014	0	0.007

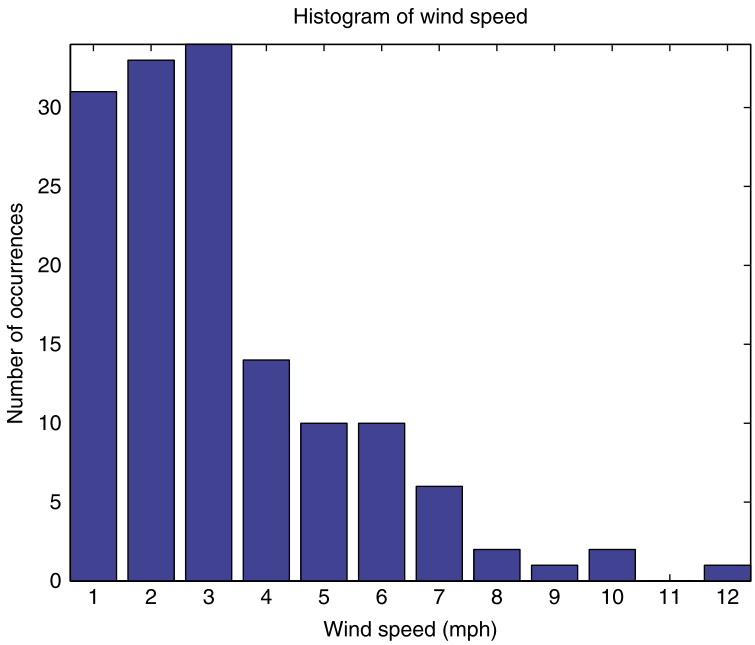


FIGURE 3.3

Histogram of synthetic wind speed observations over a 24-hour period.

that accuracy. For this situation, the observations were only to one decimal point. Therefore, we are not able to give the probability of a specific wind speed, but the probability of a wind speed being less than a stated speed.

As we can see from Fig. 3.3, there appears to be a function that models the behavior of the wind data and this function represents the probability of a value falling in a specific range. Given this function, we can then evaluate the probability by integrating the function between two points to obtain the probability of the wind speed being in this interval. This leads to the formal definition of the **probability density function (PDF)**, which is the continuous variate equivalent of the PMF as.

Definition 3.22. The probability density function (PDF), $f(x)$, of the continuous random variable, X, is a function whose integral evaluated between $x = a$ and $x = b$, where we assume that b is greater than a, gives the probability that the random variable X takes a value in the interval $[a, b]$.

A function must satisfy two very important properties to be considered as PDF, which are:

$$f(x) \geq 0, \quad \forall x, \tag{3.13a}$$

$$\int_a^b f(x)\, dx = 1, \tag{3.13b}$$

where a and b are the interval that the random variable is defined over.

As we can see from the wind example, the values do not go below zero, so the lower limit for the associated PDF would be 0. In the next section we shall present more details about distribution theory and look at the limits over which the different PDFs are defined.

As with the discrete random variable case, the associated **cumulative distribution function (CDF)** is defined as follows:

Definition 3.23. The cumulative distribution function, $F(s)$, for $a < s \leq b$ of the continuous random variable X with probability density function $f(x)$, is such that

$$F(s) = P(X \leq s) \equiv \int_a^s f(x)\, dx. \tag{3.14}$$

An important property to note here is that as the CDF is the integral of the PDF, which implies that the PDF is the derivative of the CDF.

3.4 Discrete Distribution Theory

Over the previous subsections, we have worked toward the PMF to model possible outcomes of a discrete random variable, and then toward the PDF to model the possible range of continuous random variables. In the previous section we did present a PMF, the binomial distribution. In this section we shall present a series of univariate discrete distributions that are important in different geosciences applications.

3.4.1 Binomial Distribution

We derived the binomial distribution in the previous section whose PMF is given by

$$P(R = r) = \left(\begin{array}{c} n \\ r \end{array} \right) \tau^r (1 - \tau)^{n-r},$$

where the probability, τ, must satisfy the condition $0 < \tau < 1$. If a discrete random variable is binomially distributed, then it is denoted as $X \sim B(n, \tau)$. The circumstances that a binomial distribution could be used are:

- when there are a fixed number of trials, n;
- when there are only two outcomes, effectively a success or a failure, possible at each trial;
- when the trials are independent;
- when the probability τ is constant for a success occurring at each trial; and
- when the random discrete variable is the total number of successes out of the n trials.

The binomial distribution is in the class of additive distributions. This means that if we have two independent binomially distributed random variables, $x \sim B(n, \tau)$ and $Y \sim B(m, \tau)$, then the distribution of $X + Y$ is also a binomial distribution with parameters $(n + m, \tau)$. It is not a requirement that the two random binomial distributed variables have the same probability, but some further mathematics are required to prove this property.

An application of the binomial distribution is to model the toss of a coin, but for a geoscience application, we could consider the case where we have a yes/no random variable. In [134] there was a

situation where there were observations of snow cover pixels from the Moderate Resolution Imaging Spectroradiometer (MODIS) over two different areas of the Western United States during the winter of 2006–2007. During this winter there were two major *Four Corners storms*, which are deep low pressure systems that center over the Four Corners region where the states of Colorado, Arizona, New Mexico, and Utah meet. This dynamic setup enables moisture to be transported into the Front Range of Colorado that produces an upslope feature where the moisture is lifted up against the Rockies, which can result in large snow totals in this region. In December 2006 there were two such storms only a week apart, which resulted in snow being on the ground for over 60 days in some areas of Colorado; there were also subsequent storms but not as severe. However, in [134] the authors addressed the problem of how to use data assimilation when there is a discrete observation; although they refer to snow cover observations as binary observations, these could have been approximated through a binomial distribution as their observations were compared to a comparable yes/no from the numerical model. There are currently no data assimilation systems that combine discrete PMFs with continuous PDFs, but this is an interesting possible future area of research.

The binomial distribution was first proposed by Jacob Bernoulli around 1700. The reason that we have presented this distribution is that plays an important role in the derivation of the Gaussian/normal distribution.

3.4.2 Geometric Distribution

The geometric distribution comes about from considering the situation for the number of trials until a success occurs. As in the derivation of the binomial distribution, we consider the union of the probability of the outcomes until we obtain a success. It has been shown that if each observation is independent, then this is the product of the probabilities. We start by considering the random variable S that represents "*number of failures before a success*" with $S = 1$, where if we have assumed a probability of success of τ, then the first value for S is 1. If we consider $S = 2$, then we have had one failure and one success. The probability of failure is $1 - \tau$, which means that for $S = 2$, we have the probability of the union of one failure and one success when $P(S = 2) = (1 - \tau)\tau$. We shall consider one more trial to draw a conclusion as to what the probability mass function definition is. For $S = 3$ there must have been two failures and one success. Therefore, $P(S = 3) = (1 - \tau)(1 - \tau)\tau \equiv (1 - \tau)^2 \tau$. Therefore, the definition of a geometric distribution is

Definition 3.24. A discrete random variable S is said to follow a geometric distribution if

$$P(S = s) = (1 - \tau)^{s-1} \tau, \quad s = 1, 2, 3, \ldots, \tag{3.15}$$

where $0 < \tau < 1$.

As with the binomial distribution, there are five conditions that define when a geometric distribution could be applied:

- when there is a sequence of trials;
- when there are only two outcomes, effectively a success or a failure, at each trial;
- when the trials are independent;
- when the probability τ is constant for a success occurring at each trial; and
- when the random discrete variable is the number of trials taken for a success to occur, where the successful trial is also included in that final count.

3.4.3 Poisson Distribution

Another important discrete distribution is the **Poisson Distribution**, named after the French mathematician Siméon Denis Poisson (1781–1840), discovered in 1837 as the limit of a binomial distribution when the sample size $n \to \infty$ and as $\tau \to 0$, such that their product $n\tau \equiv \lambda$ remains constant.

The starting point for the derivation of the Poisson distribution is to consider the null case, $P\,(r = 0)$ in a binomial distribution model, i.e.

$$p_0 = (1-\tau)^n \equiv \left(1 - \frac{\lambda}{n}\right)^n. \tag{3.16}$$

The next step is to allow the sample size in (3.16) to tend to infinity, which results in

$$p_0 = \lim_{n\to\infty} \left(1 - \frac{\lambda}{n}\right)^n \equiv e^{-\lambda}, \tag{3.17}$$

where we have used the formula for the exponential function as the closed form to the limit above.

If we consider the next possibility in the binomial distribution, p_1, then we have

$$p_1 = n\tau\,(1-\tau)^{n-1} \equiv \frac{n\tau}{(1-\tau)} p_0 = \frac{\lambda}{\left(1 - \frac{\lambda}{n}\right)} p_0. \tag{3.18}$$

As for the $r = 0$ case, we take the limit as $n \to \infty$ in (3.18), which results in

$$\lim_{n\to\infty} p_1 = \lim_{n\to\infty} \frac{\lambda}{\left(1 - \frac{\lambda}{n}\right)} p_0 = \lim_{n\to\infty} \frac{\lambda}{\left(1 - \frac{\lambda}{n}\right)} \lim_{n\to\infty} p_0 \equiv \lambda e^{-\lambda}. \tag{3.19}$$

To complete the derivation of the Poisson distribution, we need to prove a property of the binomial distribution that enables us to link p_r with p_{r-1} so that we can effectively set up a proof by complete induction for the general expression for all values of r. We start by considering the ratio of p_r to p_{r-1}:

$$\frac{p_r}{p_{r-1}} = \frac{\frac{n!}{r!(n-r)!}}{\frac{n!}{(r-1)!(n-r+1)!}} \frac{\tau^r\,(1-\tau)^{n-r}}{\tau^{r-1}\,(1-\tau)^{n-r+1}} = \frac{n-r+1}{r} \frac{\tau}{(1-\tau)}, \tag{3.20}$$

where we have canceled matching powers of certain terms and as well as canceling matching terms in the factorial expansions. To arrive at the relationship between p_r and p_{r-1}, we multiply both sides of (3.20) by p_{r-1}, which results in

$$p_r = \frac{n-r+1}{r} \frac{\tau}{(1-\tau)} p_{r-1}. \tag{3.21}$$

For the purpose of the derivation of the Poisson distribution, we manipulate (3.21) slightly into the form

$$p_r = \frac{1 + \frac{(1-r)}{n}}{r} \frac{n\tau}{(1-\tau)} p_{r-1} \equiv \frac{1 + \frac{(1-r)}{n}}{r} \frac{\lambda}{\left(1 - \frac{\lambda}{n}\right)} p_{r-1}. \tag{3.22}$$

As with the $r = 0$ and $r = 1$ cases, we take the limit as $n \to \infty$, which results in

$$\lim_{n\to\infty} p_r \equiv \lim_{n\to\infty} \frac{1 + \frac{(1-r)}{n}}{r} \frac{\lambda}{\left(1 - \frac{\lambda}{n}\right)} p_{r-1} = \frac{\lambda}{r} \lim_{n\to\infty} p_{r-1}. \tag{3.23}$$

Expanding the p_{r-1} term using (3.21) but now with p_{r-2} on the right-hand side results in

$$\lim_{n\to\infty} p_r = \frac{\lambda}{r} \lim_{n\to\infty} \left(\frac{n + (2-r)}{(r-1)} \frac{\tau}{(1-\tau)} p_{r-2} \right). \tag{3.24}$$

Following the same manipulations performed in (3.22), it can easily be shown that

$$\lim_{n\to\infty} p_r \equiv \frac{\lambda}{r} \frac{\lambda}{(r-1)} \lim_{n\to\infty} p_{r-2}. \tag{3.25}$$

By noticing that the factor multiplying p_{r-2} is $\frac{\lambda}{(r-1)}$, we can see that the denominator has decreased by 1 for a reduction of 1 in r. Therefore, we can extend this sequence to $r = 1$ such that

$$\lim_{n\to\infty} p_r \equiv \frac{\lambda}{r} \frac{\lambda}{(r-1)} \frac{\lambda}{(r-2)} \cdots \frac{\lambda}{2} \lambda \lim_{n\to\infty} p_0 \equiv \frac{\lambda^r}{r!} e^{-\lambda}. \tag{3.26}$$

Thus, we can define the conditions for a discrete random variable to have a Poisson distribution as follows:

Definition 3.25. A discrete random variable R is said to follow a Poisson distribution if

$$P(R = r) = \frac{\lambda^r}{r!} e^{-\lambda}, \quad r = 0, 1, 2, \ldots, \lambda > 0. \tag{3.27}$$

It should be noted that the Poisson distribution is still a discrete distribution, although it is defined for all values of r to infinity, but is determined by the parameter λ.

Below are conditions for when a Poisson distribution can be used to model an event:

- Events occur at **random** in continuous time or space.
- The probability of two events occurring simultaneously is zero. This condition is referred to as events occurring **singly**.
- Given an interval where the events are suppose to occur, and given the size of the interval, then the events occur **uniformly** such that the expected number of events in an interval is proportional to its size.
- The events occur **independently**, which implies that the probability of an event occurring in one small interval is independent of the probability of it occurring in another small interval.
- The random variable is the **number of events** that occur in an interval of a given size.

3.4.4 Discrete Uniform Distribution

As the name suggests, the probabilities associated with this distribution are the same for all values the random variable can take. Thus the PMF for this setup is as follows:

Definition 3.26. A discrete random variable T that can take values $1, 2, \ldots, K$ such that

$$P(T = t) = \begin{cases} \frac{1}{K} & t = 1, 2, \ldots, K \\ 0 & t > K \end{cases} \tag{3.28}$$

is said to follow a **discrete uniform distribution**.

Common uses for the discrete uniform distribution are to model the toss of a fair coin, or the roll of a fair die to obtain a specific number.

We now move on to consider different properties of the discrete random variables.

3.5 Expectation and Variance of Discrete Random Variables

There are important descriptive statistics of the PMFs that enable us to make decisions about the results we have obtained from using them. There are three descriptive statistics: mean, mode, and median. However, for the PMFs presented in this chapter, the derivations for the modes and the medians are quite complicated. As such, we shall focus on the mean and variance for the four discrete distributions presented.

The mean is also referred to as the expected state/value or the expectation, which shall be denoted by μ. The reason for this remark is to differentiate from the **sample mean**, which is usually denoted by $\overline{x}$, which we shall not go into detail about in this section.

The mathematical definition for the expectation of a discrete random variable is

Definition 3.27. The mean/expectation value/expectation of a discrete random variable R that can take the values r_i $(i = 1, 2, \ldots \infty)$ with probabilities p_i is defined as

$$\mu = \mathbb{E}[R] \equiv \sum_{\forall i} p_i r_i, \tag{3.29}$$

where $\mathbb{E}[\cdot]$ is the notation for the expectation operator. **Note:** The $\mathbb{E}$ and μ notations are used for both discrete and continuous random variables.

3.5.1 Mean of the Binomial Distribution

If we consider a binomial distribution with PMF

$$P(r) = \binom{n}{r} \tau^r (1 - \tau)^{n-r},$$

then evaluating the expression in (3.29) for $r = 0, 1, \ldots, n$, we have

$$\begin{aligned} \mu = E[R] \equiv \sum_{r=0}^{n} r P(r) &\equiv \sum_{r=0}^{n} r \binom{n}{r} \tau^r (1 - \tau)^{n-r}, \\ &\equiv \sum_{r=0}^{n} r \frac{n!}{r!(n-r)!} \tau^r (1 - \tau)^{n-r}. \end{aligned} \tag{3.30}$$

The first thing to recall here is that when $r = 0$, $P(r = 0) = 0$. Therefore, this term can be removed from the summation in (3.30).

Before we can continue, we must introduce the binomial series and its convergent expression. The theorem for the binomial series is as follows.

Theorem 3.28. *Let a and b be any two numbers and let n be a positive integer, then*

$$(a+b)^n = a^n + \binom{n}{1} a^{n-1}b + \binom{n}{2} a^{n-2}b^2 + \cdots + \binom{n}{n-1} ab^{n-1} + b^n. \tag{3.31}$$

Therefore, for the situation in (3.30), we have to notice that the r multiplying the fraction cancels with the r term in the r factorial component in (3.30). However, to make the summation in (3.30) match the expression above, we have to factorize out a n and a τ as well, which results in

$$\mathbb{E}[R] = n\tau \sum_{r=1}^{n} \frac{(n-1)!}{(r-1)!\,(n-r)!} \tau^{r-1} (1-\tau)^{n-r}. \tag{3.32}$$

We can see that the right-hand side of (3.32) is equivalent to the expression in (3.31) where $b \equiv \tau$, $a \equiv (1-\tau)$ and $n = n-1$. This then allows us to write the expectation of a binomial random variable as

$$\mu = n\tau\,(\tau + (1-\tau))^{n-1} \equiv n\tau. \tag{3.33}$$

3.5.2 Mean of the Geometric Distribution

We now consider how to derive the mean of a geometric distributed random variable. The starting point is the geometric distribution's PMF, given by

$$P(R) = \tau\,(1-\tau)^{r-1},$$

where $r = 1, 2, \ldots$. Therefore, recalling the definition for the expectation, we have

$$\begin{aligned} \mu = \mathbb{E}[R] &= p_1 + 2p_2 + 3p_3 + \cdots + rp_r + \cdots, \\ &= \tau + 2\tau(1-\tau) + 3\tau(1-\tau)^2 + \cdots + r\tau(1-\tau)^{r-1} + \cdots. \end{aligned} \tag{3.34}$$

The next step is to notice that we can factorize a τ from each term in (3.34). To make the next step clearer, we shall denote $q \equiv (1-\tau)$, which then enables (3.34) to be simplified as

$$\mu = \mathbb{E}[R] = \tau\left(1 + 2q + 3q^2 + \cdots + rq^{r-1} + \cdots\right). \tag{3.35}$$

To be able to find a closed expression for the summation in (3.35), we introduce the **binomial theorem**:

Theorem 3.29. *Let α be any real number, $(\alpha \in \mathbb{R})$, and also that x is a real number, $(x \in \mathbb{R})$ such that $-1 < x < 1$, then*

$$(1+x)^\alpha = 1 + \frac{\alpha}{1!}x + \frac{\alpha(\alpha-1)}{2!}x^2 + \cdots + \frac{\alpha(\alpha-1)\ldots(\alpha-r+1)}{r!}x^r + \cdots. \tag{3.36}$$

What may not be obvious is how this expression applies to the expectation in (3.35). The answer is to set $\alpha = -2$ and $x = -q$ in (3.36), which results in α minus a number term alternating in sign with the x^n terms; they are also one order of magnitude greater than the factorial term in the denominator,

which means that they cancel with all but the last term minus one. Therefore, the right-hand side of (3.35) can be shown to be equivalent to

$$(1-q)^{-2} \equiv 1 + 2q + 3q^2 + \cdots + rq^{r-1} \cdots,$$

where we have to state that the expansion is valid for q as this is equivalent to $1-\tau$, which satisfies the condition $|q| \equiv |(1-\tau)| < 1$. Thus, the expectation of a geometric distributed random variable is

$$\mu = \mathbb{E}[R] \equiv \frac{\tau}{(1-(1-\tau))^2} = \frac{\tau}{\tau^2} = \frac{1}{\tau}. \tag{3.37}$$

3.5.3 Mean of the Poisson Distribution

We now consider the mean for the Poisson distribution. We start with the definition of the Poisson distribution's PMF;

$$P(R=r) = \frac{\lambda^r}{r!} e^{-\lambda}.$$

Therefore, the expectation for a discrete random variable that follows a Poisson PMF is

$$\mu = \mathbb{E}[R] = \sum_{r=0}^{\infty} r P(r) \equiv \sum_{r=0}^{\infty} r \frac{\lambda^r}{r!} e^{-\lambda}. \tag{3.38}$$

The next step is to notice that the r cancels with the largest term in $r!$, leaving $(r-1)!$ in the denominator of (3.38). Next, we factorize $e^{-\lambda}$ out of the summation. We can also change the limits of the summation as the term at $r=0$ is equal to zero. Finally, to make the power of λ consistent with the factorial in the denominator, we factor out a λ from the summation. Thus, (3.38) can be written as

$$\mu = \mathbb{E}[R] = \lambda e^{-\lambda} \sum_{r=1}^{\infty} \frac{\lambda^{r-1}}{(r-1)!} \equiv \lambda e^{-\lambda} e^{\lambda} = \lambda, \tag{3.39}$$

where we have used the summation identity for the exponential function,

$$e^{\lambda} \equiv \sum_{i=1}^{\infty} \frac{\lambda^{i-1}}{(i-1)!}.$$

3.5.4 Mean of the Discrete Uniform Distribution

We move on to the last of the four discrete probability distributions introduced earlier; the discrete uniform distribution, whose PMF is

$$P(T=t) \equiv \begin{cases} \frac{1}{k} & t = 1, 2, \ldots, k, \\ 0 & t > k \end{cases}$$

Applying the definition for the expectation to this distribution yields

$$\mu = \mathbb{E}[T] \equiv \frac{1}{k} + \frac{2}{k} + \frac{3}{k} + \cdots + \frac{k}{k} \equiv \frac{1}{k} \sum_{t=1}^{k} t. \tag{3.40}$$

The summation on the right-hand side of (3.40) has a closed form which is given by

$$\sum_{t=1}^{k} t \equiv \frac{k\,(k+1)}{2}.$$

Thus, the expected value for a discrete uniformly distributed random variable is

$$\mu = \frac{1}{k}\frac{k\,(k+1)}{2} = \frac{k+1}{2}. \tag{3.41}$$

3.5.5 Variance of a Discrete Probability Mass Function

We now move on to calculating the variance for a discrete random variable following either a binomial, geometric, Poisson, or discrete uniform distribution. The definition for the variance, denoted by σ^2, or $Var\,[R]$, is

$$\sigma^2 \equiv Var\,[R] = \mathbb{E}\left[(X - \mathbb{E}\,[X])^2\right]. \tag{3.42}$$

Expanding out the square on the right hand side of (3.42) results in

$$\sigma^2 = \mathbb{E}\left[X^2 - 2X\mathbb{E}\,[X] + (\mathbb{E}\,[X])^2\right] \equiv \mathbb{E}\left[X^2\right] - \mathbb{E}\,[X]^2 \equiv \mathbb{E}\left[X^2\right] - \mu^2, \tag{3.43}$$

where we have used the following property of the expectation operator:

$$\mathbb{E}\,[aX] = a\mathbb{E}\,[X]. \tag{3.44}$$

For discrete PMFs, it is quite difficult to evaluate $\mathbb{E}\left[X^2\right]$; however, there is an identity that we quickly prove here that enables us to find an easier expression to evaluate. But first we introduce another property of the expectation operator. If we have two random variables X and Y and we have their sum $X + Y$, then

$$\mathbb{E}\,[X + Y] = \mathbb{E}\,[X] + \mathbb{E}\,[Y]. \tag{3.45}$$

The identity that we require is

$$\mathbb{E}\left[R^2\right] \equiv \mathbb{E}\,[R\,(R-1)] + \mathbb{E}\,[R]. \tag{3.46}$$

To prove the identity in (3.46), we expand the product inside the brackets in the first expectation, which results in

$$\mathbb{E}\left[R^2\right] \equiv \mathbb{E}\left[R^2 - R\right] + \mathbb{E}\,[R].$$

When we use the properties in (3.44) and (3.45), we have

$$\mathbb{E}\left[R^2\right] \equiv \mathbb{E}\left[R^2\right] - \mathbb{E}\,[R] + \mathbb{E}\,[R] = \mathbb{E}\left[R^2\right]. \tag{3.47}$$

Given the identity in (3.46), we variance can be expressed as

$$\sigma^2 \equiv \mathbb{E}\,[R\,(R-1)] + \mathbb{E}\,[R] - \mathbb{E}\,[R]^2. \tag{3.48}$$

3.5.6 Variance of the Binomial Distribution

We start with the mean for the binomial distribution and notice that the first term we have to derive is $\mathbb{E}[R(R-1)]$, where here this is

$$\mathbb{E}[R(R-1)] = \sum_{r=0}^{n} r(r-1)\, p_r \equiv \sum_{i=0}^{n} \frac{r(r-1)\, n!}{r!\,(n-r)!} \tau^r (1-\tau)^{n-r}. \tag{3.49}$$

The first feature to note is that the terms for $r = 0$ and $r = 1$ in the summation in (3.49) are both zero. Therefore, the summation starts at $r = 2$. We manipulate the terms in the summation that are not dependent on r so that the terms in the summation resemble a binomial series; as such

$$\begin{aligned}\mathbb{E}[R(R-1)] &= n(n-1)\tau^2 \sum_{r=2}^{n} \frac{(n-2)!}{r!\,(n-r)!} \tau^{r-2} (1-\tau)^{n-r}, \\ &= n(n-1)\tau^2 (\tau + (1-\tau))^{n-2}, \\ &= n(n-1)\tau^2. \end{aligned} \tag{3.50}$$

The final step to arrive at the variance of a binomial distribution is to evaluate (3.48) using (3.50);

$$\begin{aligned}\sigma^2 &\equiv \mathbb{E}[R(R-1)] + \mathbb{E}[R] - \mathbb{E}[R]^2, \\ &= n(n-1)\tau^2 + n\tau - n^2\tau^2, \\ &= n\tau - n\tau^2 = n\tau(1-\tau). \end{aligned} \tag{3.51}$$

3.5.7 Variance of the Geometric Distribution

We now move on the geometric distributed discrete random variables. However, for this distribution the derivation is not as straightforward as for the binomial distribution, or for the mean of the geometric PMF. Before we start, we require three properties of geometric series. The reason will become clear when we start the derivation for $E[R(R-1)]$.

A geometric series is defined as

$$g(r) = \sum_{k=0}^{\infty} ar^k \equiv a + ar + ar^2 + ar^3 + ar^4 + \cdots \equiv \frac{a}{(1-r)} \equiv a(1-r)^{-1}, \tag{3.52}$$

where $|r| < 1$, and r in (3.52) is referred to as the **common ratio**. The next step is to differentiate (3.52) with respect to r to obtain

$$g'(r) \equiv \frac{d}{dr}\left(\sum_{k=0}^{\infty} ar^k\right) \equiv 0 + a + 2ar + 3ar^2 + 4ar^3 + \cdots \equiv \frac{a}{(1-r)^2} \equiv a(1-r)^{-2}. \tag{3.53}$$

The third property of the geometric series involves the second derivative of (3.52) with respect to r, which is:

$$g''(r) \equiv \frac{d^2}{dr^2}\left(\sum_{k=0}^{\infty} ar^k\right) \equiv 0 + 0 + 2a + 6ar + 12ar^2 + \cdots \equiv \frac{2a}{(1-r)^3} \equiv a(1-r)^{-3}. \tag{3.54}$$

It should be noted that the differentiated series in (3.53) and (3.54) are equivalent to

$$\frac{d}{dr}\left(\sum_{k=0}^{\infty} ar^k\right) \equiv \sum_{k=1}^{\infty} kar^{k-1}, \tag{3.55a}$$

$$\frac{d^2}{dr^2}\left(\sum_{k=0}^{\infty} ar^k\right) \equiv \sum_{k=2}^{\infty} k\,(k-1)\,ar^{k-2}. \tag{3.55b}$$

Returning to the variance of a geometrically distributed discrete random variable, we require $\mathbb{E}[R(R-1)]$, which is

$$\begin{aligned} E[R(R-1)] &\equiv 1(1-1)p_1 + 2(2-1)p_2 + 3(3-1)p_3 + 4(4-1)p_4 + \cdots + r(r-1)p_r + \cdots, \\ &\equiv 2\tau(1-\tau) + 6\tau(1-\tau)^2 + 12\tau(1-\tau)^3 + \cdots + r(r-1)\tau(1-\tau)^{r-1} + \cdots, \\ &\equiv \sum_{r=1}^{\infty} r(r-1)\tau(1-\tau)^{r-1}. \end{aligned} \tag{3.56}$$

We can see that the series in (3.56) is not quite in the form as that of the series in (3.55b); however, this can be remedied by factoring out $\tau(1-\tau)$, which makes (3.56)

$$E[R(R-1)] = \tau(1-\tau)\sum_{r=2}^{\infty} r(r-1)(1-\tau)^{k-2} \equiv \frac{2\tau(1-\tau)}{(1-(1-\tau))^3} = \frac{2(1-\tau)}{\tau^2}. \tag{3.57}$$

Recalling that variance is defined as $\sigma^2 = Var[R] \equiv \mathbb{E}[R(R-1)] + \mathbb{E}[R] - \mathbb{E}[R]^2$, implies that the variance of a discrete random variable from a geometric distribution is

$$\sigma^2 = Var[R] = \frac{2(1-\tau)}{\tau^2} + \frac{1}{\tau} - \frac{1}{\tau^2} = \frac{2-2\tau+1+\tau}{\tau^2} = \frac{1-\tau}{\tau^2}. \tag{3.58}$$

3.5.8 Variance of the Poisson Distribution

We now consider the variance of a discrete random variable that follows a Poisson distribution. As before, the first step is to derive the $\mathbb{E}[R(R-1)]$ term, which is

$$\begin{aligned} \mathbb{E}[R(R-1)] &= \sum_{r=0}^{\infty} r(r-1)P(r) \equiv \sum_{r=0}^{\infty} r(r-1)\frac{\lambda^r}{r!}e^{-\lambda} = e^{-\lambda}\sum_{r=2}^{\infty}\frac{\lambda^r}{(r-2)!} = \lambda^2 e^{-\lambda}\sum_{r=2}^{\infty}\frac{\lambda^{r-2}}{(r-2)!}, \\ &= \lambda^2 e^{-\lambda}e^{\lambda} = \lambda^2. \end{aligned} \tag{3.59}$$

The reason why the sum starts from $r = 2$ is because the terms for $r = 0$ and $r = 1$ are zero.

Therefore, we have that the variance for the Poisson distribution is

$$\sigma^2 = Var[R] \equiv \lambda^2 + \lambda - (\lambda)^2 = \lambda. \tag{3.60}$$

The expression in (3.60) is a very important property of the Poisson distribution. It demonstrates that the mean and the variance are the same and this property is often used to verify if their data are approximately following a Poisson distribution.

3.5.9 Variance of the Discrete Uniform Distribution

We finish with the variances of discrete random variables that follow a discrete uniform distribution. It is more convenient to use the $Var[R] = E\left[R^2\right] - (E[R])^2$ definition here. However, before we start, we state an expression for the arithmetic sum of squares, which is

$$\sum_{k=1}^{N} k^2 = \frac{N(N+1)(2N+1)}{6}. \tag{3.61}$$

Therefore, the variance of a discrete uniformly distributed random variable is

$$\begin{aligned}\sigma^2 = Var[R] \equiv \sum_{r=1}^{K} \frac{1}{K} r^2 - \left(\sum_{r=1}^{K} \frac{1}{K} r\right)^2 &= \frac{(K+1)(2K+1)}{6} - \frac{(K+1)^2}{4}, \\ = (K+1)\left(\frac{2(2K+1) - 3(K+1)}{12}\right) &= \frac{(K+1)(K-1)}{12} \equiv \frac{K^2-1}{12}.\end{aligned} \tag{3.62}$$

Quite often in statistical analysis, and in data assimilation, another summary statistic you will see is the **standard deviation**. This is often abbreviated to st. dev. This summary statistic is simply the square root of the variance and is denoted by σ.

3.6 Moments and Moment-Generating Functions

Most PMFs can be defined by what are referred to as their *moments*. The first moment of a PMF is the mean, which we have already derived for the binomial, geometric, Poisson, and discrete uniform distributed discrete random variables.

However, the second moment of a PMF is not the variance as such. There are two type of moments; the first set are the **uncentered moments**, often denoted as μ'_m for the mth moment, and defined by

Definition 3.30. The mth **uncentered moment**, μ'_m, of a discrete random variable with PMF, $P(r)$, is

$$\mu'_m = \mathbb{E}\left[R^m\right] = \sum_{\forall r} r^m p_r. \tag{3.63}$$

As we can see from the definition in (3.63), the variance as defined before is not the second moment.

The second set of moments are the **central moments** and are usually denoted by μ_m. The definition for central moments is as follows.

Definition 3.31. The mth **central moment**, μ_m, of a discrete random variable with a PMF $P(r)$ is

$$\mu_m = \mathbb{E}\left[(R-\mu)^m\right] \equiv \mathbb{E}\left[(R - \mathbb{E}[R])^m\right] = \sum_{\forall r} (r-\mu)^m p_r. \tag{3.64}$$

Therefore, we can see from (3.64) that the variance is the **second central moment** of a probability mass function.

While the first and second central moments are important (mean and variances), the third and fourth order central moments are part of the definition of the **standardized moments** that are often critical in

identifying a distribution. The third order standardized moment is referred to as the **skewness** and is a measure of how far away a distribution is from symmetry. Non-symmetric distributions can be either left or right skewed. A distribution that is symmetric about its mean will have a zero skewness value.

The fourth order standardized moment of a PMF is referred to as the coefficient of **kurtosis**, which is a measure of the *peakedness* of a distribution.

The definition for skewness, often denoted by γ_1, in terms of expected values, is

$$\gamma_1 \equiv \mathbb{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] \equiv \frac{\mathbb{E}\left[(X-\mathbb{E}[X])^3\right]}{\left(\mathbb{E}\left[(X-\mathbb{E}[X])^2\right]\right)^{\frac{3}{2}}} \equiv \frac{\mu_3}{\sigma^3}. \tag{3.65}$$

Expanding the cubic expression results in

$$\begin{aligned}
\gamma_1 &\equiv \frac{\mathbb{E}\left((X-\mathbb{E}[X])\,(X-\mathbb{E}[X])^2\right]}{\sigma^3}, \\
&= \frac{\mathbb{E}\left[(X-\mathbb{E}[X])\left(X^2-2X\mathbb{E}[X]+(\mathbb{E}[X])^2\right)\right]}{\sigma^3}, \\
&= \frac{\mathbb{E}\left[X^3-3X^2\mathbb{E}[X]+3X\,(\mathbb{E}[X])^2-(\mathbb{E}[X])^3\right]}{\sigma^3}, \\
&= \frac{\mathbb{E}\left[X^3\right]-3\mathbb{E}\left[X^2\right]\mathbb{E}[X]+3\,(\mathbb{E}[X])^3-(\mathbb{E}[X])^3}{\sigma^3}, \\
\gamma_1 &= \frac{\mathbb{E}\left[X^3\right]-3\mu\mathbb{E}\left[X^2\right]+2\mu^3}{\sigma^3}.
\end{aligned} \tag{3.66}$$

However, if we do not cancel the μ^3 terms, then we can write the skewness in terms of the mean and the variance as

$$\begin{aligned}
\gamma_1 &= \frac{\mathbb{E}\left[X^3\right]-3\mathbb{E}[X]\left(\mathbb{E}\left[X^2\right]-(\mathbb{E}[X])^2\right)-(\mathbb{E}[X])^3}{\sigma^3}, \\
&= \frac{\mathbb{E}\left[X^3\right]-3\mu\sigma^2-\mu^3}{\sigma^3}.
\end{aligned} \tag{3.67}$$

The definition for kurtosis involves a similar ratio between different orders of central moments. Kurtosis involves the ratio of the fourth central moment to the square of the second central moment, or rather the square of the variance, or the quartic of the standard deviation. However, we should note that there are two different definitions for kurtosis; where both are provided below.

$$\beta_2 \equiv \mathbb{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] \equiv \frac{\mathbb{E}\left[(X-\mathbb{E}[X])^4\right]}{\left(\mathbb{E}\left[(X-\mathbb{E}[X])^2\right]\right)^2} \equiv \frac{\mu_4}{\sigma^4}. \tag{3.68}$$

As with the derivation of the skewness, we expand the numerator in (3.68) to obtain the expression for kurtosis in terms of the other moments, which is

$$\begin{aligned}\beta_2 &\equiv \frac{\mathbb{E}\left[\left(X - 2\mathbb{E}[X]X + (\mathbb{E}[X])^2\right)\left(X - 2\mathbb{E}[X]X + (\mathbb{E}[X])^2\right)\right]}{\sigma^4}, \\ &= \frac{\mathbb{E}\left[X^4 - 4\mathbb{E}[X]X^3 + 6(\mathbb{E}[X])^2X^2 - 4(\mathbb{E}[X])^3X + (\mathbb{E}[X])^4\right]}{\sigma^4}, \\ &= \frac{\mathbb{E}\left[X^4\right] - 4\mu\mathbb{E}\left[X^3\right] + 6\mu^2\mathbb{E}\left[X^2\right] - 4\mu^3\mathbb{E}[X] + \mu^4}{\sigma^4}. \end{aligned} \tag{3.69}$$

The expression in (3.69) is the most common expression for kurtosis; however, if you have already calculated the skewness, variance, and mean, then it would be quicker, and more cost-efficient, to have an expression for kurtosis in terms of these lower order central moments. This can be done by rearranging the definitions for the skewness and variance to have an expression for $\mathbb{E}\left[X^3\right]$ and $\mathbb{E}\left[X^2\right]$ that are functions of skewness, variance, and mean, and variance and mean respectively. These expressions are

$$\mathbb{E}\left[X^3\right] = \gamma_1\sigma^3 + 3\mu\sigma^2 + \mu^3, \tag{3.70}$$

$$\mathbb{E}\left[X^2\right] = \sigma^2 + \mu^2. \tag{3.71}$$

Applying (3.70) and (3.71) in (3.69) results in the expression for kurtosis in terms of skewness, variance, and mean as

$$\beta_2 = \frac{\mathbb{E}\left[X^4\right] - 4\gamma_1\sigma^3\mu - 6\sigma^2\mu^2 - \mu^4}{\sigma^4}. \tag{3.72}$$

The second definition for kurtosis, often referred to as the **excess kurtosis**, denoted as γ_2, is similar to the definition in (3.68), but now we have

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3. \tag{3.73}$$

The -3 term comes about because the normal/Gaussian distribution, which we shall introduce later in this chapter, does not have a zero value for kurtosis using (3.68). The actual value for kurtosis of a Gaussian distribution is 3, and hence this number is subtracted off.

However, as we have seen when deriving the variance of the various discrete distributions, it can be difficult to find the expectation of the powers of the discrete random variables, so we now introduce what are referred to as the **moment-generating functions** to help address this problem.

3.6.1 Moment-Generating Functions for Probability Mass Functions

The moment-generating function can be used for both discrete and continuous random variables, when a closed form for the function exists. For discrete random variables, the moment-generating function is defined as

$$M(t) = \mathbb{E}\left[e^t\right] \equiv \sum_{\forall r} p_r e^{rt}. \tag{3.74}$$

While the expression in (3.74) does not look any easier, we need to recall that the exponential can be expressed in terms of a sum and so (3.74) becomes

$$M(t) = \sum_{\forall r} \left(1 + rt + \frac{(rt)^2}{2!} + \frac{(rt)^3}{3!} + \cdots\right) p_r, \tag{3.75}$$

where if we multiply throughout by p_r, and take the summation through, then we can rewrite (3.75) as

$$M(t) = \sum_{\forall r} p_r + \sum_{\forall r} rtp_r + \sum_{\forall r} \frac{(rt)^2}{2!} p_r + \sum_{\forall r} \frac{(rt)^3}{3!} p_r + \sum_{\forall r} \frac{(rt)^4}{4!} p_r + \cdots. \tag{3.76}$$

It should be clear that the expression in (3.76) is similar to the definitions for the non-central moments from the last section, but also that the first sum in (3.76) is equal to one, as this is the definition for the CDF for a discrete PMF. Therefore, we have that the moment-generating function can be expressed as

$$\begin{aligned} M(t) &= 1 + t\mu + \frac{t^2}{2!}\mu_2' + \frac{t^3}{3!} + \mu_3' + \frac{t^4}{4!}\mu_4' + \cdots + \frac{t^n}{n!}\mu_n' + \cdots, \\ &= 1 + t\mathbb{E}[R] + \frac{t^2}{2!}\mathbb{E}\left[R^2\right] + \frac{t^3}{3!}\mathbb{E}\left[R^3\right] + \frac{t^4}{4!}\mathbb{E}\left[R^4\right] + \cdots + \frac{t^n}{n!}\mathbb{E}\left[R^n\right] + \cdots. \end{aligned} \tag{3.77}$$

However, we still have the problem of how to extract out the specific non-central moment from (3.77). The answer to this question has already indirectly presented itself in the derivation of the variance for the Poisson distribution. In that derivation, we differentiated the geometric series, and its summed expression, to separate out the expression for $\mathbb{E}\left[R^2\right]$. Therefore, if we differentiate (3.77) with respect to t and set $t = 0$, then we see that the non-central moments corresponds to the equivalent order derivative of the moment-generating function when t is set to zero. For example

$$\begin{aligned} M(t)|_{t=0} &= 1, \\ \mathbb{E}[R] \equiv \left.\frac{dM(t)}{dt}\right|_{t=0} &= \mu, \\ \mathbb{E}\left[R^2\right] \equiv \left.\frac{d^2M(t)}{dt^2}\right|_{t=0} &= \mu_2', \\ \mathbb{E}\left[R^3\right] \equiv \left.\frac{d^3M(t)}{dt^3}\right|_{t=0} &= \mu_3', \\ \mathbb{E}\left[R^4\right] \equiv \left.\frac{d^4M(t)}{dt^4}\right|_{t=0} &= \mu_4'. \end{aligned}$$

In summary, the nth non-central moment can be expressed in terms of the nth order derivative with respect to t of the moment-generating function as

$$E\left[R^n\right] \equiv \left.\frac{d^nM(t)}{dt^n}\right|_{t=0}. \tag{3.78}$$

To complete this section, we now derive the moment-generating functions for the binomial, geometric, Poisson, and discrete uniform distributions.

3.6.2 Binomial Distribution Moment-Generating Function

For the remainder of this section, we shall abbreviate moment-generating function to MGF. Therefore, given the definitions for the MGF and the PMF binomial distribution, then we require

$$M_{Bi}(t) = \sum_{r=1}^{n} e^{rt} p_r \equiv \sum_{r=1}^{n} e^{rt} \binom{n}{r} \tau^r (1-\tau)^{n-r} \equiv \sum_{r=1}^{n} e^{rt} \frac{n!}{r!(n-r)!} \tau^r (1-\tau)^{n-r}. \tag{3.79}$$

Collecting the terms that are powers of r, we can rewrite (3.79) as

$$M_{Bi}(t) = \sum_{r=0}^{n} \binom{n}{r} \left(\tau e^t\right)^r (1-\tau)^{n-r}. \tag{3.80}$$

The sum in (3.80) is a geometric distribution with $a = (1-\tau)$ and $b = \tau e^r$. Thus the summation in (3.80) enables us to define the binomial distribution's MGF as

$$M_{Bi}(t) = \left((1-\tau) + \tau e^t\right)^n. \tag{3.81}$$

To verify the statement in the last section that the order of the derivatives of the MGFs match the corresponding order of the non-central moments of the distribution, we shall differentiate (3.81) with respect to t. The first derivative of (3.81) is

$$\mu = \left.\frac{dM_{Bi}}{dt}\right|_{t=0} = n\left((1-\tau) + \tau e^t\right)^{n-1} \tau e^t \Big|_{t=0} = n\left((1-\tau) + \tau\right)^{n-1} \tau = n\tau. \tag{3.82}$$

To find the second non-central moment, we now take the second derivative of (3.81) evaluated at $t = 0$, which is

$$\begin{aligned} \mu_2' = \left.\frac{d^2 M_{Bi}}{dt^2}\right|_{t=0} &= n(n-1)\left((1-\tau) + \tau e^t\right)^{n-2} \tau^2 e^{2t} + n\left((1-\tau) + \tau e^t\right)^{n-1} \tau e^t \Big|_{t=0}, \\ &= n(n-1)\tau^2 + n\tau. \end{aligned}$$

To verify the variance for a binomially distributed discrete random variable, we require $\sigma^2 = \mu_2' - \mu^2$:

$$\sigma_{Bi}^2 = n(n-1)\tau^2 + n\tau - n^2\tau^2 = n\tau(1-\tau).$$

3.6.3 Geometric Distribution Moment-Generating Function

As for the binomial distribution, we require $E\left[e^{RT}\right]$ but now evaluated for the geometric distribution. Therefore, the geometric distribution's MGF, $M_{Ge}(t)$, is defined as

$$M_{Ge}(t) \equiv \sum_{r=1}^{\infty} e^{rt} \tau (1-\tau)^{r-1} = \tau e^t \sum_{r=1}^{\infty} e^{(r-1)t} (1-\tau)^{r-1} = \tau e^t \sum_{r=1}^{\infty} \left((1-\tau) e^t\right)^{r-1}. \tag{3.83}$$

The next step is to recall the closed form for a geometric series, under the condition that the common ratio is less than one, (3.52); applying the summation closed form from (3.52) to (3.83) enables us to

define the MGF for the geometric distribution as

$$M_{Ge}(t) = \frac{\tau e^t}{\left(1-(1-\tau)e^t\right)}. \tag{3.84}$$

We now need to determine for what values of t the geometric series is convergent. A geometric session is dependent on the common ratio being less than 1. Therefore, we require

$$\begin{aligned}\left|(1-\tau)e^t\right| < 1 \Rightarrow e^t < \frac{1}{(1-\tau)} \Rightarrow t < \ln 1 - \ln(1-\tau),\\ \Rightarrow t < -\ln(1-\tau).\end{aligned}$$

To verify that the MGF presented in (3.84) is that for the geometric distribution, we shall show that the first and second derivatives of (3.84) are the same as the mean and the non-central second moment derived earlier. Therefore, the first derivative of (3.84) is

$$\begin{aligned}\mu_{ge} = \left.\frac{dM_{ge}}{dt}\right|_{t=0} &= \left.\frac{\tau e^t\left(1-(1-\tau)e^t\right) - \tau e^t\left(-(1-\tau)e^t\right)}{\left(1-(1-\tau)e^t\right)^2}\right|_{t=0},\\ &= \left.\frac{\tau e^t - \tau(1-\tau)e^{2t} + \tau(1-\tau)e^{2t}}{\left(1-(1-\tau)e^t\right)^2}\right|_{t=0} = \frac{\tau}{\tau^2} = \frac{1}{\tau}.\end{aligned}$$

Verifying that the non-central second moment is the second derivative of M_{ge} we have

$$\begin{aligned}\mu'_{2,ge} = \left.\frac{d^2M_{ge}}{dt^2}\right|_{t=0} &\equiv \frac{d}{dt}\left(\frac{\tau e^t}{\left(1-(1-\tau)e^t\right)^2}\right)\Bigg|_{t=0},\\ &= \left.\frac{\left(1-(1-\tau)e^t\right)^2\tau e^t + 2\left(1-(1-\tau)e^t\right)(1-\tau)e^t\tau e^t}{\left(1-(1-\tau)e^t\right)^4}\right|_{t=0},\\ &= \left.\frac{\left(1-(1-\tau)e^t\right)\tau e^t + 2\left((1-\tau)e^t\right)\tau e^t}{\left(1-(1-\tau)e^t\right)^3}\right|_{t=0},\\ &= \left.\frac{\tau e^t\left(1+(1-\tau)e^t\right)}{\left(1-(1-\tau)e^t\right)^3}\right|_{t=0} = \frac{\tau(1+(1-\tau))}{\tau^3},\\ &= \frac{2-\tau}{\tau^2}.\end{aligned}$$

Using the expression above, we can verify that we obtain the same expression for the variance of the geometric distribution to that we derived earlier as

$$\sigma^2_{ge} = \mu'_{2,ge} - \mu^2_{ge} = \frac{2-\tau}{\tau^2} - \frac{1}{\tau^2} = \frac{1-\tau}{\tau^2}.$$

3.6.4 Poisson Moment-Generating Function

The moment-generating function for the Poisson distribution is quite easy to derive compared to the last two derivations:

$$M_P(t) = \mathbb{E}\left[e^{Rt}\right] \equiv \sum_{r=0}^{\infty} e^{Rt} P(R) \equiv \sum_{r=0}^{\infty} e^{rt} \frac{e^{-\lambda}\lambda^r}{r!}, \equiv e^{-\lambda} \sum_{r=0}^{\infty} \frac{e^{rt}\lambda^r}{r!},$$
$$= e^{-\lambda} \sum_{r=0}^{\infty} \frac{\left(e^t\lambda\right)^r}{r!}. \tag{3.85}$$

The next step is to notice that the sum in (3.85) is equivalent to the power series of the exponential function acting on $e^t\lambda$. Therefore, the Poisson MGF is

$$M_P(t) = e^{-\lambda} e^{e^t\lambda} \equiv e^{\lambda\left(e^t-1\right)}. \tag{3.86}$$

As with the previous two distributions, we now verify the relationship to the first two non-central moments. Therefore, the first derivative with respect to t of (3.86) is

$$\mu_P \equiv \left.\frac{dM_P}{dt}\right|_{t=0} = \left.e^{\lambda\left(e^t-1\right)}\lambda e^t\right|_{t=0} = \lambda e^0 e^{\lambda\left(e^0-1\right)} = \lambda.$$

The second derivative of (3.86) with respect to t is

$$\mu'_{2,P} \equiv \left.\frac{d^2M_P}{dt^2}\right|_{t=0} = \left.\lambda e^t e^{\lambda\left(e^t-1\right)} + \left(\lambda e^t\right)\left(\lambda e^t\right) e^{\lambda\left(e^t-1\right)}\right|_{t=0},$$
$$= \left.\left(\lambda^2 e^{2t} + \lambda e^t\right) e^{\lambda\left(e^t-1\right)}\right|_{t=0},$$
$$= \lambda^2 + \lambda.$$

To complete this subsection, we verify that the variance, calculated as $\sigma_P^2 = \mu'_{2,P} - \mu_P^2$, is equal to λ:

$$\sigma_P^2 = \lambda^2 + \lambda - \lambda^2 = \lambda.$$

3.6.5 Discrete Uniform Distribution Moment-Generating Function

The MGF for the discrete uniform distribution is not straightforward to derive, as it is quite different than for the more complicated defined Poisson distribution. However, the full derivation is as follows:

$$M_{du}(t) \equiv \sum_{r=1} K \frac{1}{K} e^{rt} = \frac{1}{K} \sum_{r=0}^{K} e^{rt}. \tag{3.87}$$

While the expression in (3.87) looks easy to evaluate, we require a closed form for the sum of the exponentials. The first step is to define the sum as $S = \sum_{r=1}^{K} e^{rt}$ and then expand the sum to show all the terms up to K To start we denote $\omega = e^t$, and so we have

$$S = \sum_{r=1}^{K} \left(e^t\right)^r \equiv \omega + \omega^2 + \omega^3 + \cdots + \omega^K,$$
$$= \omega\left(1 + \omega + \omega^2 + \cdots + \omega^{K-1}\right). \tag{3.88}$$

The next step is to multiply both sides of (3.88) by ω:

$$\omega S = \omega^2 \left(1 + \omega + \omega^2 + \cdots + \omega^{K-1}\right). \tag{3.89}$$

Taking the difference between (3.88) and (3.89) and factorizing results in

$$(1-\omega)\, S = \omega - \omega^{K+1} \Rightarrow S = \frac{\omega - \omega^{K+1}}{(1-\omega)} \equiv \frac{e^t \left(1 - e^{Kt}\right)}{1 - e^t}. \tag{3.90}$$

However, we have to be careful here; we have divided by a term that can be **zero**, and so we must determine under what conditions will the denominator be equal to zero. It is clear that the denominator will only be equal to zero when $t = 0$. Therefore, the discrete uniform's MGF is defined as

$$M_{du}(t) = \begin{cases} \frac{\left(1-e^{Kt}\right)}{K\left(e^{-t}-1\right)} & t \neq 0 \\ 1 & t = 0 \end{cases}, \tag{3.91}$$

where the value for $t = 0$ is obtained by l'Hôpital's rule.

The question now becomes how can we differentiate (3.91) to obtain the non-central moments, as the function is not differentiable at $t = 0$. The answer is we take the derivatives and then take the limit as $t \to 0$.

3.7 Continuous Distribution Theory

We now move on to the continuous probability density functions. There are many different forms of PDFs that are used for different aspects of geophysical modeling. In this section we shall present the properties of the different distributions in a different way to how we presented them for the discrete PMFs. In this section we shall present each distribution one after the other where we shall present its moment-generating function, if it exists, along with the moments, medians (if they exist), and their modes. As we progress to the extreme value distributions, we shall encounter some quite advanced mathematics, but we shall present this in a way that we hope is clear to follow. We start with the most used distribution, and the basis for many of the current operational data assimilation systems: the Gaussian (normal) distribution.

3.7.1 Gaussian (Normal) Distribution

This continuous distribution has two names, and depending on the scientific field that is using it, it is referred to as the Gaussian or the normal distribution. In engineering, meteorological, and oceanic fields the distribution is referred to as Gaussian. However, in mathematical and statistical communities it is referred to as the normal distribution. Throughout this book we shall be using the Gaussian terminology when we refer to its use after Gauss's discovery.

There are conflicting accounts of when the Gaussian distribution arose. As the name "Gaussian" suggests, the mathematician Carl Friedreich Gauss (1777–1855) is credited with its discovery. However, the first derivation of the normal distribution occurred before Gauss and was derived by Abraham de Moivre (1667–1754) around 1733 as an approximation to the binomial distribution. Besides Gauss, the

Gaussian distribution's discovery is also associated with Pierre-Simon Laplace (1749–1827). It should be noted that Gauss's and Laplace's association with the normal distribution arises from weighted least square minimization for residuals. It appears that the name "normal" came about due to the usage of the terminology by the British statistician Karl Pearson (1857–1936) and is very well associated with different mathematical and statistical techniques and terms that are important in data assimilation.

Here we start with de Moivre's contribution in discovering the Gaussian distribution. If we consider the four plots in Fig. 3.4, where each plot is of the binomial distribution with different sample sizes $n = 2, 4, 8, 16$, we see what appears to be a smooth continuous function starting to form as the sample size increases. The structure of the curve appears to be approximating a "bell," and quite often the Gaussian curve is referred to as the *bell curve*. Therefore, if we let n tend to infinity, what would the result be? The answer is the Gaussian distribution. We now present the proof of the Gaussian distribution as the limit of the binomial distribution as $n \to \infty$; this is the proof to De Moivre-Laplace theorem, where it is assumed that the PMF of the random number of successes observed in a series of n independent Bernoulli trials, each having probability p of success, is therefore a binomial distribution with n trials, converging to the PDF of the Gaussian distribution with mean np and standard deviation $\sqrt{np(1-p)}$, as n grows large, and where it is assumed that $p \neq 0$ or 1.

Theorem 3.32. ***De Moivre-Laplace Theorem****: As n grows large, for r in the neighborhood of* $n\tau$*, then it is possible to approximate*

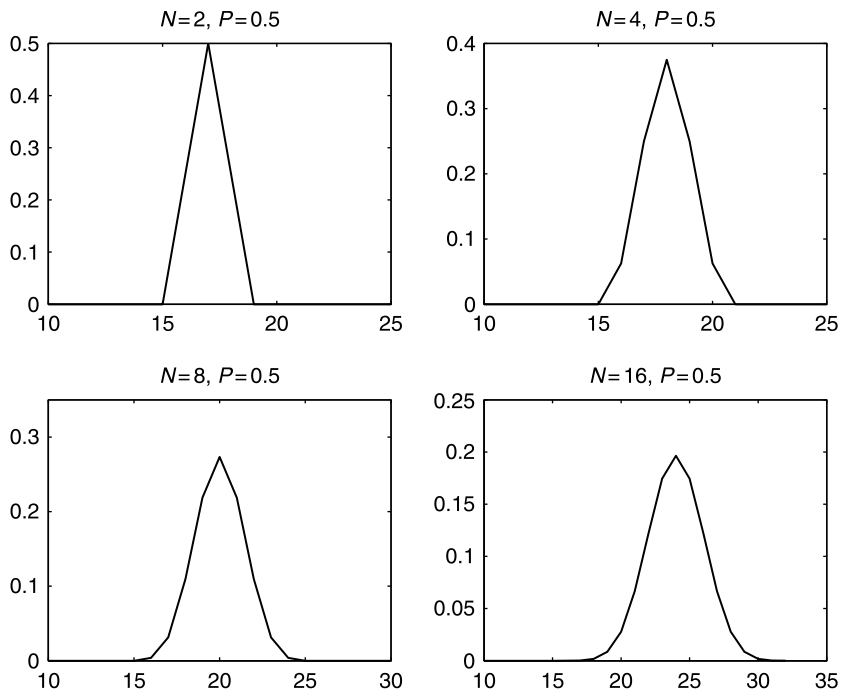


FIGURE 3.4

Illustration of the binomial distribution tending toward the Gaussian distribution as n increases.

$$\binom{n}{r}\tau^r(1-\tau)^{n-r} \sim \frac{1}{\sqrt{2\pi n\tau(1-\tau)}}\exp\left\{-\frac{1}{2}\frac{r-n\tau}{n\tau(1-\tau)}\right\}, \quad p\tau > 0, \tag{3.92}$$

in the sense that the ratio of the left-hand side of (3.92) *to the right-hand side converges to* 1 *as* $n \to \infty$.

Proof. The first property that needs to be observed here is that r cannot be fixed, otherwise it would quickly fall outside the range of interest as $n \to \infty$. Therefore, what is required is that r be allowed to vary but always to be a fixed number of standard deviations away from the mean. The effect of this is to make these values of r always be associated with the same point on what is referred to as the ***standard Gaussian distribution***. This is achieved through defining $r = n\tau + x\sqrt{n\tau(1-\tau)}$ for some fixed x. Taking the example of setting $x = 1$, then r will always be one standard deviation away from the mean. As a result of this definition for r, we obtain two approximations as $n \to \infty$: $r \to n\tau$ and $\frac{r}{n} \to \tau$.

The path that the proof of the De Moivre-Laplace theorem takes is to transform the left-hand side of (3.92) in to the right-hand side. The transformation is achieved by using three sets of approximations. The first approximation is referred to as **Stirling's Formula**, which is given by the following.

Definition 3.33. Stirling's Formula:

$$n! \approx n^n e^{-n}\sqrt{2\pi n}, \quad n \to \infty. \tag{3.93}$$

Now, applying Stirling's Formula to the binomial distribution results in

$$\begin{aligned}\binom{n}{r}\tau^r(1-\tau)^{n-r} &= \frac{n!}{r!(n-r)!}\tau^r(1-\tau)^{n-r},\\ &= \frac{n^n e^{-n}\sqrt{2\pi n}}{r^r e^{-r}\sqrt{2\pi r}(n-r)^{n-r}e^{-(n-r)}\sqrt{2\pi(n-r)}}\tau^r(1-\tau)^{n-r},\\ &= \sqrt{\frac{n}{2\pi r(n-r)}}\left(\frac{n\tau}{r}\right)^r\left(\frac{n(1-\tau)}{n-r}\right)^{n-r}.\end{aligned} \tag{3.94}$$

The next assumption is that $\frac{r}{n} \to \tau$, this enables the matching of the first term on the right-hand side of (3.92):

$$\begin{aligned}\binom{n}{r}\tau^r(1-\tau)^{n-r} &\approx \sqrt{\frac{1}{2\pi n\frac{r}{n}\left(1-\frac{r}{n}\right)}}\left(\frac{n\tau}{r}\right)^r\left(\frac{n(1-\tau)}{n-r}\right)^{n-r},\\ &= \frac{1}{\sqrt{2\pi n\tau(1-\tau)}}\left(\frac{n\tau}{r}\right)^r\left(\frac{n(1-\tau)}{n-r}\right)^{n-r}.\end{aligned} \tag{3.95}$$

The third and final approximation we require is the Taylor series approximation to $\ln(1+x)$, given by

$$\ln(1+x) \approx x - \frac{x^2}{2} + \frac{x^3}{3} - \cdots,$$

along with the property that the exponential is the inverse of the logarithm. Thus

$$\begin{aligned}&\binom{n}{r}\tau^r(1-\tau)^{n-r}\\ &\approx \frac{1}{\sqrt{2\pi n\tau(1-\tau)}}\exp\left\{\ln\left(\frac{n\tau}{r}\right)^r + \ln\left(\frac{n(1-\tau)}{n-r}\right)^{n-r}\right\},\end{aligned}$$

$$\approx \frac{1}{\sqrt{2\pi n\tau\,(1-\tau)}} \exp\left\{-r\ln\left(\frac{r}{n\tau}\right) - (r-n)\ln\left(\frac{n-r}{n\,(1-\tau)}\right)\right\},$$

$$\approx \frac{1}{\sqrt{2\pi n\tau\,(1-\tau)}} \exp\left\{-r\ln\left(\frac{n\tau + x\sqrt{n\tau\,(1-\tau)}}{n\tau}\right) + (r-n)\ln\left(\frac{n-n\tau - x\sqrt{n\tau\,(1-\tau)}}{n\,(1-\tau)}\right)\right\},$$

$$\approx \frac{1}{\sqrt{2\pi n\tau\,(1-\tau)}} \exp\left\{-r\ln\left(1 + x\sqrt{\frac{(1-\tau)}{n\tau}}\right) + (r-n)\ln\left(1 - x\sqrt{\frac{\tau}{n\,(1-\tau)}}\right)\right\}. \tag{3.96}$$

To make the derivation easier to read, given that the terms in (3.96) are to be approximated by Taylor series expansion for $\ln(1+x)$, we let $\upsilon = 1-\tau$ such that $\tau + \upsilon = 1$. We also introduce the constant, C, such that $C = \frac{1}{\sqrt{2\pi n\tau\upsilon}}$. Therefore, substituting the Taylor series for $\ln(1+x)$ into (3.96), we obtain:

$$\begin{aligned}
&\binom{n}{r}\tau^r\upsilon^{n-r} \\
&\approx C\exp\left\{-r\left(x\sqrt{\frac{\upsilon}{n\tau}} - \frac{x^2\upsilon}{2n\tau} + \cdots\right) + (r-n)\left(-x\sqrt{\frac{\tau}{n\upsilon}} - \frac{x^2\tau}{2n\upsilon} - \cdots\right)\right\}, \\
&\approx C\exp\left\{\left(np - x\sqrt{n\tau\upsilon}\right)\left(x\sqrt{\frac{\upsilon}{n\tau}} - \frac{x^2\upsilon}{2n\tau} + \cdots\right)\right. \\
&\qquad\qquad \left. -\left(n\upsilon - x\sqrt{n\tau\upsilon}\right)\left(-x\sqrt{\frac{\tau}{n\upsilon}} - \frac{x^2\tau}{2n\upsilon} - \cdots\right)\right\}, \\
&\approx C\exp\left\{\left(-x\sqrt{n\tau\upsilon} + \frac{1}{2}x^2\upsilon - x^2\upsilon + \cdots\right) + \left(x\sqrt{n\tau\upsilon} + \frac{1}{2}x^2\tau + x^2\tau - \cdots\right)\right\}, \\
&\approx C\exp\left\{-\frac{1}{2}x^2\upsilon - \frac{1}{2}x^2\tau - \cdots\right\}, \\
&= C\exp\left\{-\frac{1}{2}x^2\upsilon - \frac{1}{2}x^2(\tau+\upsilon)\right\}.
\end{aligned} \tag{3.97}$$

It was stated earlier that τ and υ have the property $\tau + \upsilon = 1$. Therefore, substituting for C it is possible to write (3.97) as

$$\binom{n}{r}\tau^r\upsilon^{n-r} \approx \frac{1}{\sqrt{2\pi n\tau\,(1-\tau)}}\exp\left\{-\frac{x^2}{2}\right\} \equiv \frac{1}{\sqrt{2\pi n\tau\,(1-\tau)}}\exp\left\{-\frac{1}{2}\frac{(r-n\tau)^2}{n\tau\,(1-\tau)}\right\}.$$

An important feature to recall here is that the mean and variance of a binomial random variable have been shown to be $E(r) = n\tau$ and $VAR = n\tau\,(1-\tau)$, respectively. Therefore, looking at the expression above, we can see that the Gaussian distribution is defined by the square of the random variable minus the mean and then divided by its variance.

The expression for the Gaussian distribution above in between the approximation sign and the equivalence sign is referred to as the **standardized normal distribution**, as it is centered around zero with unit variance. We can therefore give the definition of the standardized Gaussian distribution as

$$f(x) = \frac{1}{\sqrt{2\pi}}\exp\left\{-\frac{1}{2}x^2\right\}, \quad x \in (-\infty, \infty), \tag{3.98}$$

that has the following properties:

- The graph of the standardized normal distribution, as seen in Fig. 3.5, is symmetrical about the origin.
- As stated in (3.98), the values that the continuous random variable x can take are between negative infinity and infinity.
- The maximum of the standardized Gaussian curve occurs at the origin and it can easily be shown that the value at this point is $(2\pi)^{-\frac{1}{2}} = 0.3989$.
- There are two points of inflexion of the curve which occur at $x = -1$ and $x = 1$.

So while the derivation of the proof of the De Moivre-Laplace theorem above shows that the Gaussian distribution is the limit as $n \to \infty$ of the binomial distribution, it does not explain where the Gaussian distribution come from. It assumes that it is already known, i.e., the right-hand side of the approximation sign. The De Moivre-Laplace theorem is in fact a special case of the **central limit theorem**, which we shall go into more detail later. We now present the derivation of the Gaussian distribution from a geometric and calculus viewpoint.

Calculus-based derivation of the Gaussian distribution

The starting point in the calculus-based derivation of the Gaussian distribution is to consider a person throwing a projectile at the origin of the Cartesian plane. The person is aiming at the origin, but random errors in their throws will produce varying results. Given this situation, the following three assumptions are made:

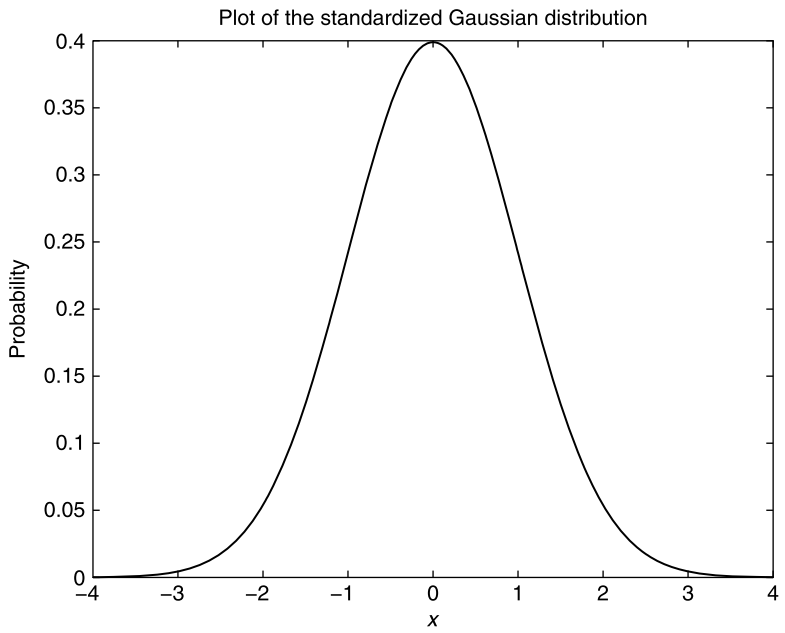


FIGURE 3.5

Plot of the standardized Gaussian distribution.

1. The errors do not depend on the orientation of the coordinate system.
2. Errors that are in perpendicular directions are independent. This implies that throwing too high does not alter the probability of being off to the right.
3. Large errors are less likely than small errors.

To determine the shape of the distribution, we consider the probability of the projectile falling in a vertical strip from x to $x + \Delta x$, and denote this probability as $p(x)\,\Delta x$. Similarly, let the probability of the projectile landing in the horizontal strip from y to $y + \Delta y$ be $p(y)\,\Delta y$. The intersection of these two strips creates a boxed area that the projectile could fall into, which has an associated probability $p(x)\,\Delta x \times p(y)\,\Delta y$. Since it was assumed that the orientation does not matter, any region that is r units away from the origin with the area $\Delta x \times \Delta y$ has the same probability. Therefore, it is possible to say that

$$p(x)\,\Delta x \times p(y)\,\Delta y = g(r)\,\Delta x \Delta y \Rightarrow g(r) = p(x)\,p(y). \tag{3.99}$$

Differentiating both sides of (3.99) with respect to θ yields

$$0 = p(x)\frac{dp(y)}{d\theta} + p(y)\frac{dp(x)}{d\theta}, \tag{3.100}$$

since g is independent of the orientation, and therefore θ.

The next step is to introduce the parametric change of variables $x = r\cos\theta$ and $y = r\sin\theta$. This allows (3.100) to be written as

$$0 = p(x)\,p'(y)\,r\cos\theta - p(y)\,p'(x)\,r\sin\theta, \tag{3.101}$$

where the $'$ indicates differentiating with respect to x or y, accordingly. Substituting x and y back into (3.101), and rearranging, results in the differential equation,

$$\frac{p'(x)}{xp(x)} = \frac{p'(y)}{yp(y)}, \tag{3.102}$$

which can be solved through the technique referred to as **separating the variables**.

The differential equation in (3.102) is true for any x and y, assuming that x and y are independent. This can only be true if the ratio defined by the differential equation in (3.102) is a constant. Therefore,

$$\frac{p'(x)}{xp(x)} = \frac{p'(y)}{yp(y)} = C.$$

Solving for $p(x)$ yields

$$\frac{p'(x)}{p(x)} = Cx,$$

which is the gradient of a logarithm acting on a function. Therefore, integrating both sides of the expression above results in

$$\ln p(x) = \frac{Cx^2}{2} + d. \tag{3.103}$$

Applying the exponential to both sides of (3.103) and recognizing that the gradient is true up to some constant yields

$$p(x) = A \exp\left\{\frac{C}{2}x^2\right\}. \tag{3.104}$$

Since it has been assumed that large errors are less likely to occur than small errors, it is implied that C must be negative. Therefore, the final general form for $p(x)$ is

$$p(x) = A \exp\left\{-\frac{k}{2}x^2\right\}, \quad k > 0, \tag{3.105}$$

which is the basic form for the Gaussian distribution. The next step is to determine the two constants A and k.

First we start with A; it has already been stated that one of the conditions for a function of a discrete random variable to be considered as a PMF is that it must sum to one over all of its values. The equivalent condition for a continuous PDF is given by

$$c(x) = \int_a^b p(x)\, dx = 1, \quad x \in (a, b), \tag{3.106}$$

where $c(x)$ is the cumulative density function.

Therefore, applying (3.106) to (3.105), where it is assumed for this function $x \in (-\infty, \infty)$, we obtain

$$\int_{-\infty}^{\infty} A \exp\left\{-k\frac{x^2}{2}\right\} dx = 1, \quad x \in (-\infty, \infty).$$

Rearranging the expression above, and noticing that the function is symmetric, results in

$$\int_0^{\infty} \exp\left\{-k\frac{x^2}{2}\right\} dx = \frac{1}{2A}, \quad x \in [0, \infty).$$

Combining with a second dummy variable, y, defined by the same function and over the same interval, results in

$$\left(\int_0^{\infty} \exp\left\{-k\frac{x^2}{2}\right\} dx\right)\left(\int_0^{\infty} \exp\left\{-k\frac{y^2}{2}\right\} dy\right) = \frac{1}{4A^2}, \quad x \in [0, \infty), y \in [0, \infty).$$

Recalling that we assumed that x and y are independent, enables the product of the two integrals to be written as the double integral

$$\int_0^{\infty}\int_0^{\infty} \exp\left\{-\frac{k}{2}\left(x^2 + y^2\right)\right\} dy\, dx = \frac{1}{4A^2}, \quad x \in [0, \infty), y \in [0, \infty). \tag{3.107}$$

Introducing polar coordinates to evaluate the double integral enables (3.107) to be rewritten as

$$\int_0^{\infty}\int_0^{\infty} \exp\left\{-\frac{k}{2}\left(x^2 + y^2\right)\right\} dy\, dx \equiv \int_0^{\frac{\pi}{2}}\int_0^{\infty} \exp\left\{-\frac{k}{2}r^2\right\} r\, dr\, d\theta, \quad r\,[0, \infty), \theta \in \left[0, \frac{\pi}{2}\right]. \tag{3.108}$$

There is an improper integral in (3.108), and as such a u-substitution is required, where $u = -\frac{kr^2}{2} \Rightarrow du = -krdr$ for $u \in [0, -\infty)$, for the integral to be evaluated in polar coordinates. Therefore the evaluation of the integrals in (3.108) become

$$\int_0^{\frac{\pi}{2}} \int_0^{\infty} \exp\left\{-\frac{k}{2}r^2\right\} r\, dr\, d\theta \equiv \int_0^{\frac{\pi}{2}} -\frac{1}{k}\left[\int_0^{-\infty} \exp\{u\}\, du\right] d\theta = \int_0^{\frac{\pi}{2}} \frac{d\theta}{k} = \frac{\pi}{2k}. \tag{3.109}$$

This then implies that $\frac{1}{4A^2} = \frac{\pi}{2k}$. Rearranging yields that $A = \sqrt{\frac{k}{2\pi}}$ and therefore the probability density function becomes

$$p(x) = \sqrt{\frac{k}{2\pi}} \exp\left\{-\frac{k}{2}x^2\right\}.$$

To be able to find the value for k, we introduce the continuous definitions for the mean and variance.

Definition 3.34. The mean, μ, of a continuous random variable, x, for a probability density function, $p(x)$, is defined as

$$\mu \equiv \mathbb{E}[X] = \int_a^b xp(x)\, dx, \quad x \in [a, b], \tag{3.110}$$

where $\mathbb{E}[\cdot]$ is the expectation operator for a continuous random variable.

Definition 3.35. The variance, σ^2, of a continuous random variable with PDF, $p(x)$ is defined as

$$\sigma^2 \equiv \mathbb{E}\left[(X-\mu)^2\right] = \int_a^b (x-\mu)^2 p(x)\, dx, \quad x \in [a, b]. \tag{3.111}$$

Returning to the derivation of the Gaussian PDF. If we now consider (3.110), then we have that $xp(x)$ is an odd function, which implies that its mean is zero. Now we recall the variance definition from (3.111),

$$\sigma^2 = 2\sqrt{\frac{k}{2\pi}} \int_0^{\infty} x^2 \exp\left\{-\frac{k}{2}x^2\right\} dx, \quad x \in [0, \infty).$$

To evaluate this integral, we apply integration by parts, where $u = x$ and $v = x\exp\left\{-\frac{k}{2}x^2\right\}$, integrate between $x = 0$ and $x = M$, and then let $M \to \infty$, which results in

$$\sigma^2 = 2\sqrt{\frac{k}{2\pi}} \left(\left[\lim_{M\to\infty} -\frac{x}{k}\exp\left\{-\frac{k}{2}x^2\right\}\right]_0^M + \frac{1}{k}\int_0^{\infty} \exp\left\{-\frac{k}{2}x^2\right\} dx\right). \tag{3.112}$$

Taking the limit of $M \to \infty$ makes the first term in (3.112) equal to zero. We already know the value of the integral in (3.112), as it was used to prove the value of A to ensure the cumulative density function added up to one. Therefore, we have that the variance is equal to

$$\sigma^2 = 2\sqrt{\frac{k}{2\pi}}\frac{1}{k}\int_0^{\infty} \exp\left\{-\frac{k}{2}x^2\right\} dx = 2\frac{\sqrt{k}}{\sqrt{2\pi}}\frac{1}{k}\frac{\sqrt{2\pi}}{2\sqrt{k}} \Rightarrow k = \frac{1}{\sigma^2}. \tag{3.113}$$

Now we have the expression for a version of Gaussian distribution, denoted by $G(0, \sigma)$, given by

$$G\left(0, \sigma^2\right) \equiv \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{x^2}{2\sigma^2}\right\}, \quad x \in (-\infty, \infty).$$

The general form of the Gaussian distribution can be shown to be

$$G\left(\mu, \sigma^2\right) \equiv \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}, \quad x \in (-\infty, \infty). \tag{3.114}$$

The first expression above for the Gaussian distribution is almost equivalent to the standard normal distribution. The difference is that the standard Gaussian distribution has a mean of zero and a variance of one. Therefore, the standard Gaussian distribution, $G(0, 1)$, is given by

$$G(0, 1) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{x^2}{2}\right\}, \quad x \in (-\infty, \infty).$$

The CDF for the Gaussian distribution is quite often denoted by $\Phi(b)$ and is defined by

$$\Phi(b) = \int_{-\infty}^{b} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{x^2}{2}\right\} dx. \tag{3.115}$$

The integral in (3.115) cannot be explicitly evaluated; however, it is possible to evaluate it numerically by using quadrature approximations. We shall not do that here, but the values for general points of the standard Gaussian distribution can be found in most introductory statistics textbooks.

An important feature to note about the standard Gaussian distribution is that the area under the curve between $x = -1$ and $x = 1$ is 68% of the total area under the curve. Therefore 68% of the probability is concentrated between $x = -1$ and $x = 1$. The percentage of the total area under the curve between $x = -2$ and $x = 2$ is 95%, where between $x = -3$ and $x = 3$ it is 99.7%. For $x \in (-\infty, \infty)$, there is only 0.3% of the distribution that is distant more than ± 3 units.

To generalize this fact, the units are actually standard deviations, σ, away from the mean, where the distribution is centered. Therefore, it is possible to say that 99.7% of the probability of the Gaussian distribution is within $\mu \pm 3\sigma$. This is an important feature of the Gaussian distribution and will become important in observational quality control later.

Before moving on to the moments of the Gaussian distribution, we consider the effects that the two parameters, μ and σ^2, have on the shape of the distribution. We can see from Fig. 3.6 that by changing the value of μ but keeping the variance equal to one, the shape of the distribution remains the same but is centered at the values for μ.

In Fig. 3.7 we have kept $\mu = 0$ for all three plots, but now allow σ^2 to be less than one and greater than one. We can see the effect that the variance has on the shape of the distribution, where for $\sigma^2 > 1$, the distribution starts to spread along the x axis, while when $\sigma^2 < 1$, the distribution becomes narrower and only covers a few values of the x axis. Therefore, the value used for σ^2 determines the spread of the distribution along the x axis.

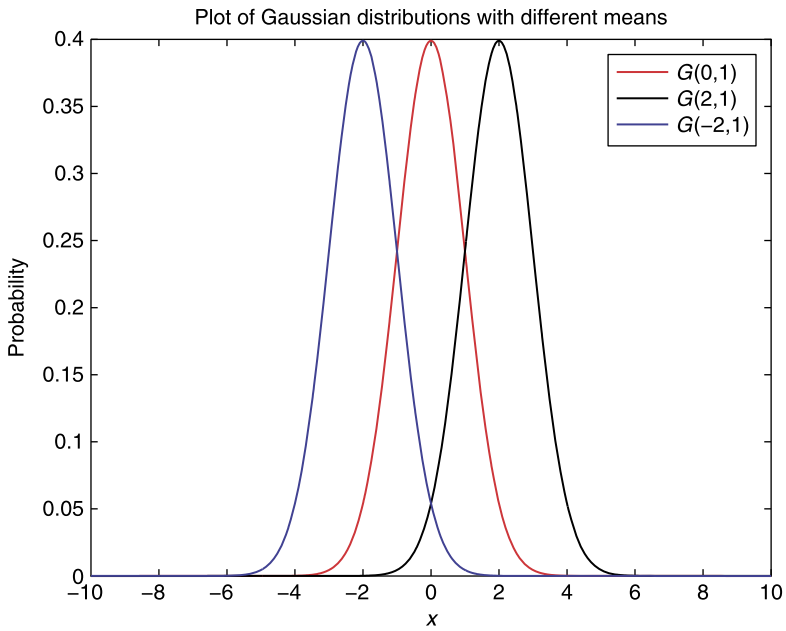


FIGURE 3.6

Plots of Gaussian distribution with unit variance but changing means.

3.7.2 Moments of the Gaussian Distribution

In deriving the Gaussian distribution we introduced the definitions to derive the mean and variance for a continuous random variable. We now apply these definition to the general version of the Gaussian distribution, but first we present three identities for some of the integrals involved in their derivation.

$$\int_{-\infty}^{\infty} \exp\left\{-\frac{1}{2}x^2\right\} dx = \sqrt{2\pi},$$
$$\int_{-\infty}^{\infty} x \exp\left\{-\frac{1}{2}x^2\right\} dx = 0,$$
$$\int_{-\infty}^{\infty} x^2 \exp\left\{-\frac{1}{2}x^2\right\} dx = \sqrt{2\pi}.$$

Recalling the definition for the continuous mean, we have

$$\mathbb{E}[X] = \frac{1}{\sigma\sqrt{2\pi}} \int_{-\infty}^{\infty} x \exp\left\{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right\} dx, \quad x \in (-\infty, \infty).$$

We now introduce the change of variable, $t = \frac{x-\mu}{\sigma}$, which is equivalent to $x = \mu + \sigma t$. This then gives $dx = \sigma\, dt$ and the limits of integration stay the same. Therefore, the integral above can be written in

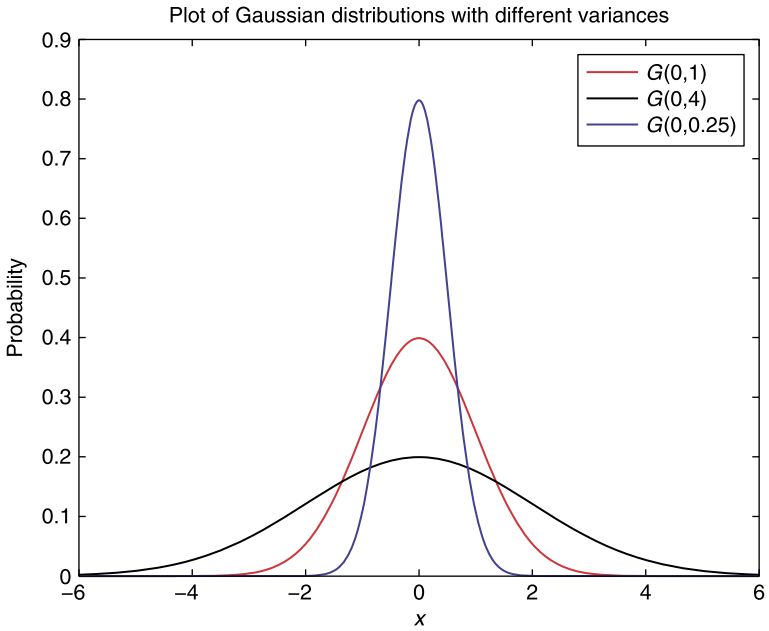


FIGURE 3.7

Plots of Gaussian distribution with zero mean but with changing variances.

terms of t as

$$\begin{aligned}\mathbb{E}[X] &= \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{\infty}\sigma\,(\mu+\sigma t)\exp\left\{-\frac{t^2}{2}\right\}dt, \quad t\in(-\infty,\infty),\\ &= \frac{\mu}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\exp\left\{-\frac{t^2}{2}\right\}dt + \frac{\sigma}{\sqrt{2\pi}}\int_{-\infty}^{\infty}t\exp\left\{-\frac{t^2}{2}\right\}dt,\\ &= \frac{\mu}{\sqrt{2\pi}}\sqrt{2\pi} + \frac{\sigma}{\sqrt{2\pi}}0,\\ &= \mu.\end{aligned}$$

The definition for the variance applied to the general Gaussian distribution is

$$Var[X] \equiv \mathbb{E}\left[(X-\mu)^2\right] \equiv \int_{-\infty}^{\infty}(x-\mu)^2\exp\left\{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right\}dx, \quad x\in(-\infty,\infty).$$

Again, introducing the same change of variable as used for the derivation of the mean yields

$$Var[X] = \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{\infty}\sigma^3 t^2\exp\left\{-\frac{t^2}{2}\right\}dt, \quad t\in(-\infty,\infty),$$

$$
\begin{aligned}
&= \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^2 \exp\left\{-\frac{t^2}{2}\right\} dt, \\
&= \frac{\sigma^2}{\sqrt{2\pi}} \sqrt{2\pi}, \\
&= \sigma^2.
\end{aligned}
$$

Therefore, the mean and the variance of the general Gaussian distribution are μ and σ^2, respectively.

We move on to consider the next two standardized moments which are defined in the same manner as for the discrete PMF: skewness and kurtosis. The definitions for the continuous versions of skewness and kurtosis are given by the following.

Definition 3.36. The skewness of a continuous probability density function $p(x)$ is given by

$$
\begin{aligned}
\beta_1 &\equiv \mathbb{E}\left[\left(\frac{X - \mathbb{E}[X]}{\left(\mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2\right)^{\frac{1}{2}}}\right)^3\right], \\
&= \frac{\int_a^b x^3 p(x)\,dx - 3\int_a^b x p(x)\,dx \int_a^b (x-\mu)^2 p(x)\,dx - \left(\int_a^b x p(x)\,dx\right)^3}{\left(\int_a^b (x-\mu)^2 p(x)\,dx\right)^{\frac{3}{2}}}, \\
&\equiv \frac{\int_a^b x^3 p(x)\,dx - 3\mu\sigma^2 - \mu^3}{\sigma^3}.
\end{aligned} \tag{3.116}
$$

Definition 3.37. The kurtosis of a continuous probability density function, $p(x)$, is given by

$$
\begin{aligned}
\gamma_2 &\equiv \mathbb{E}\left[\left(\frac{X - \mathbb{E}[X]}{\left(\mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2\right)^{\frac{1}{2}}}\right)^4\right], \\
&= \frac{\int_a^b x^4 p(x)\,dx - 4\int_a^b x p(x)\,dx \int_a^b x^3 p(x)\,dx + 6\left(\int_a^b x p(x)\,dx\right)^2 \int_a^b x^2 p(x)\,dx - 3\left(\int_a^b x p(x)\,dx\right)^4}{\left(\int_a^b (x-\mu)^2 p(x)\,dx\right)^2}, \\
&= \frac{\int_a^b x^4 p(x)\,dx - 4\mu\int_a^b x^3 p(x)\,dx + 6\mu^2 \int_a^b x^2 p(x)\,dx - 3\mu^4}{\sigma^4}, \\
&= \frac{\mathbb{E}\left[X^4\right] - 4\mu\mathbb{E}\left[X^3\right] + 6\mu^2\mathbb{E}\left[X^2\right] - 3\mu^4}{\sigma^4}.
\end{aligned} \tag{3.117}
$$

We now derive the skewness for the Gaussian distribution. We start by introducing the same change of variable used for the mean and variance, noticing that

$$
\mathbb{E}\left[x^3\right] = \mu^3 + 3\mu^2\sigma t + 3\mu\sigma^2 t^2 + \sigma^3 t^3.
$$

From the derivation of the variance, we know that the term multiplying μ^3 integrates to $\sqrt{2\pi}$, the term multiplying t integrates to 0, and the term multiplying $3\mu\sigma^2$ also integrates to $\sqrt{2\pi}$. Therefore, the last term to consider is the one multiplying t^3.

We need to integrate $\frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^3 \exp\left\{-\frac{t^2}{2}\right\} dx$. The trick is again to manipulate the integral through integration by parts, where here we have

$$u = t^2, \qquad v' = t \exp\left\{-\frac{t^2}{2}\right\},$$
$$u' = 2t, \qquad v = -\exp\left\{-\frac{t^2}{2}\right\},$$

which makes the integral above

$$\left[-t^2 \exp\left\{-\frac{t^2}{2}\right\}\right]_{-\infty}^{\infty} + 2\int_{-\infty}^{\infty} t \exp\left\{-\frac{t^2}{2}\right\} dx, \tag{3.118}$$

but both terms in (3.118) are zero and hence t^3 also integrates to zero. Therefore, the final expression for the skewness is

$$\beta_1 = \frac{\mu^3 + 3\mu\sigma^2 - 3\mu\sigma^2 - \mu^3}{\sigma^3} = 0. \tag{3.119}$$

Thus, the Gaussian distribution has zero skewness, which is expected for symmetric distributions. This is also an important property of the distribution when trying to determine which distribution our data/errors are following.

Moving on to the kurtosis, we first acknowledge that it was not possible to express this property in terms of central moments. Therefore, we introduce expressions for the second and third non-central moments of the Gaussian distribution, and leave their proofs as an exercise:

$$\mathbb{E}\left[X^2\right] = \mu^2 + \sigma^2,$$
$$\mathbb{E}\left[X^3\right] = \mu^3 + 3\mu\sigma^2.$$

However, we still need to find an expression for the fourth non-central moment of the Gaussian distribution. Introducing the same change of variable as before and expanding $(\mu + \sigma t)^4$ results in

$$(\mu + \sigma t)^4 = \mu^4 + 4\mu^3\sigma t + 6\mu^2\sigma^2 t^2 + 4\mu\sigma^3 t^3 + \sigma^4 t^4. \tag{3.120}$$

As with the third moment, we can eliminate the odd powers of t and that the t^2 term will integrate to 1; therefore the only term left to consider is the t^4. As what has become a noticeable pattern, this integral is going to be manipulated through integration by parts

$$u = t^3, \qquad v' = t \exp\left\{-\frac{t^2}{2}\right\},$$
$$u' = 3t^2, \qquad v = -\exp\left\{-\frac{t^2}{2}\right\},$$

which implies

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} t^4 \exp\left\{-\frac{1}{2}t^2\right\} dx \equiv \frac{1}{\sqrt{2\pi}}\left(\left[-t^3 \exp\left\{-\frac{t^2}{2}\right\}\right]_{-\infty}^{\infty} + 3\int_{-\infty}^{\infty} t^2 \exp\left\{-\frac{t^2}{2}\right\} dx\right). \tag{3.121}$$

The first term on the right-hand side of the equation in (3.121) is zero, and it has been shown that the second term is equal to $\sqrt{2\pi}$. Therefore, the integral is equal to 3. Thus the fourth non-central moment of the Gaussian distribution is

$$E\left[X^4\right] = \mu^4 + 6\mu^2\sigma^2 + 3\sigma^4.$$

Combining all the expressions for the first four moments of the Gaussian distribution, we have that the kurtosis of the Gaussian distribution is

$$\gamma_2 = \frac{\mu^4 + 6\mu^2\sigma^2 + 3\sigma^4 - 4\mu^4 - 12\mu^2\sigma^2 + 6\mu^4 + 6\mu^2\sigma^2 - 3\mu^4}{\sigma^4} = \frac{3\sigma^4}{\sigma^4} = 3. \tag{3.122}$$

An important feature to notice here is that the kurtosis is not equal to zero. It is often stated that the higher-order standardized moments of a Gaussian distribution above second order are zero. As such, the kurtosis definition is often redefined as

$$\hat{\gamma}_2 \equiv \gamma_2 - 3.$$

Exercise 3.38. *Show that the second and third non-central moments of the Gaussian distribution are*

$$\mathbb{E}\left[X^2\right] = \mu^2 + \sigma^2,$$
$$\mathbb{E}\left[X^3\right] = \mu^3 + 3\mu\sigma^2.$$

3.7.3 Moment-Generating Functions for Continuous Probability Density Functions

As with the discrete case, it is possible to define the moment-generating function, if it exists, for continuous PDFs. It has just been shown that the expression to calculate the non-central moments is

$$\mathbb{E}\left[x^n\right] = \int_a^b x^n f(x)\, dx,$$

which can be quite difficult to calculate.

The definition for the MGF for continuous PDFs is given by

$$m_x(t) = \mathbb{E}\left[\exp\{tx\}\right] \equiv \int_a^b \exp\{tx\} f(x)\, dx. \tag{3.123}$$

Through using the Taylor series expansion for the exponential function we have

$$\exp\{xt\} = 1 + tx + \frac{t^2x^2}{2!} + \frac{t^3x^3}{3!}\cdots. \tag{3.124}$$

Substituting (3.124) into (3.123), we obtain

$$m_x(t) = \int_a^b \left(1 + tx + \frac{t^2x^2}{2!} + \frac{t^3x^3}{3!}\cdots\right) f(x)\,dx, \tag{3.125}$$

which becomes

$$\begin{aligned} m_x(t) &= \int_a^b f(x)\,dx + t\int_a^b xf(x)\,dx + \frac{t^2}{2!}\int_a^b x^2 f(x)\,dx + \frac{t^3}{3!}\int_a^b x^3 f(x)\,dx\cdots, \\ &= 1 + t\mathbb{E}[x] + \frac{t^2}{2!}\mathbb{E}\left[x^2\right] + \frac{t^3}{3!}\mathbb{E}\left[x^3\right] + \cdots. \end{aligned}$$

Therefore, the relationship to the non-central moments is

$$\begin{aligned} m_x(t)|_{t=0} &= m_x(0) = 1, \\ \left.\frac{dm_x(t)}{dt}\right|_{t=0} &= m_x'(0) = \mathbb{E}[X], \\ \left.\frac{d^2m_x(t)}{dt^2}\right|_{t=0} &= m_x''(0) = \mathbb{E}\left[X^2\right], \end{aligned}$$

which is similar to the result for the MGF for discrete PMF.

While it may appear quite daunting to find the MGF for a specific continuous PDF, we attempt to alleviate this fear by deriving the associated function for the Gaussian distribution.

$$m_G(t) = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty} \exp\{tx\}\exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}dx. \tag{3.126}$$

We now need to complete the square of

$$\frac{-x^2 + 2\left(\mu + \sigma^2 t\right) - \mu}{2\sigma^2},$$

which becomes

$$-\frac{1}{2}\left(\frac{\left(x - \left(\mu + \sigma^2 t\right)\right)^2}{\sigma^2}\right) + \mu t + \frac{\sigma^2 t^2}{2}.$$

This then makes (3.126)

$$\begin{aligned} m_N(t) &= \exp\left\{\mu t + \frac{\sigma^2 t^2}{2}\right\}\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}\exp\left\{-\frac{1}{2}\frac{\left(x - \left(\mu + \sigma^2 t\right)\right)^2}{\sigma^2}\right\}dx, \\ m_N(t) &= \exp\left\{\mu t + \frac{\sigma^2 t^2}{2}\right\}, \end{aligned} \tag{3.127}$$

where the integral on the right-hand side is the CDF for the Gaussian distribution, $G\left(\mu + \sigma^2 t, \sigma^2\right)$, which integrates to one.

Taking the first two derivatives of (3.127) to verify that, we obtain the first two non-central moments of the Gaussian distribution, we see that

$$\begin{aligned}\left.\frac{dm_N(t)}{dt}\right|_{t=0} &= \left(\mu+\sigma^2 t\right)\exp\left\{\mu t+\frac{\sigma^2 t^2}{2}\right\}\Bigg|_{t=0},\\ &= \mu.\\ \left.\frac{d^2 m_x(t)}{dt^2}\right|_{t=0} &= \left(\sigma^2+\left(\mu+\sigma^2 t\right)\left(\mu+\sigma^2 t\right)\right)\exp\left\{\mu t+\frac{\sigma^2 t^2}{2}\right\}\Bigg|_{t=0},\\ &= \sigma^2+\mu^2.\end{aligned}$$

There are some useful properties of MGF that are required later to prove some important properties for data assimilation.

Properties of moment-generating functions

- Let a and b be constants, and let $M_X(t)$ be the moment-generating function for a continuous random variable X, then the moment-generating function of the random variable $Y = a + bX$ is

$$\begin{aligned}M_Y(t) &= E\left[e^{tY}\right] = E\left[e^{t(a+bX)}\right]\\ &= e^{at}E\left[e^{(bt)X}\right] = e^{at}M_X(bt).\end{aligned} \tag{3.128}$$

- Let X and Y be independent continuous random variables with associated moment-generating functions $M_x(t)$ and $M_Y(t)$, respectively. We then use the property that $E[g_1(X)g_2(Y)] = E[g_1(X)]E[g_2(Y)]$ for functions g_1 and g_2. Let $Z = X + Y$ be another continuous random variable, then its associated moment-generating function is $M_Z(t)$:

$$\begin{aligned}M_Z(t) &= E\left[E^{tZ}\right] = E\left[e^{t(X+Y)}\right]\\ &= E\left[e^{tX}\right]E\left[e^{tY}\right] = M_X(t)M_Y(t).\end{aligned} \tag{3.129}$$

This leads to a very important theorem that underlies the relationship between the distribution of the *errors* and that of the distributions of the true state and the background state in data assimilation.

Theorem 3.39. ***Uniqueness Theorem****: Suppose that random variables X and Y have moment-generating functions given by $m_X(t)$ and $m_Y(t)$, respectively. If $m_X(t) = m_Y(t)$ for all values of t, then X and Y have the same distribution.*

This theorem is important for a proof that will be provided when we define *errors* and detection methods to determine their distributions for data assimilation.

An important property of Gaussian distributed variables is that if X_1 and X_2 are independently Gaussian distributed random variables, with means μ_1 and μ_2, and variances σ_1^2 and σ_2^2, respectively, then their sum $X_1 + X_2$ is also a Gaussian distributed random variable such that $X_1 + X_2 \sim G\left(\mu_1+\mu_2, \sigma_1^2+\sigma_2^2\right)$. This can be generalized to N independently distributed Gaussian random variables such that the distribution of the sum is

$$\sum_{i=1}^{N} X_i \sim G\left(\sum_{i=1}^{N}\mu_i, \sum_{i=1}^{N}\sigma_i^2\right).$$

There is a similar property for the distribution of the difference between independently Gaussian distributed random variables, $X_1 - X_2$, such that the difference is also a Gaussian random variable that follows the Gaussian distribution $G\left(\mu_1 - \mu_2, \sigma_1^2 + \sigma_2^2\right)$. This property is one of the foundations of all Gaussian distribution-based data assimilation systems.

A very important theorem associated with the Gaussian distribution is the **central limit theorem**, which is stated as follows.

Theorem 3.40. ***Central Limit Theorem****: Let* $\{X_1, X_2, \ldots, X_n\}$ *be a sequence of independently and identically distributed random variables with*

$$\mathbb{E}[X_i] = \mu < \infty,$$
$$VAR[X_i] = \sigma^2 < \infty,$$

for $i = 1, 2, \ldots, n$*. Then as* n *becomes large, the distribution of the mean of the sequence* $\bar{X} = \frac{1}{n}\sum_{i=1}^{N} X_i$ *is approximately Gaussian distributed, i.e.,* $\bar{X} \sim G\left(\mu, \frac{\sigma^2}{n}\right)$*. Also the sample sum* $S = \sum_{i=1}^{N} X_i$ *is a Gaussian distributed random variable, i.e.,* $S \sim G\left(n\mu, n\sigma^2\right)$*.*

3.7.4 Median of the Gaussian Distribution

At the beginning of this chapter, we mentioned that there were three different descriptive statistics. For the discrete probability mass functions, we only presented the mean of the distribution. This statistic represents the **minimum variance estimator**. The remaining two descriptive statistics are the median and the mode. We shall now define the univariate median for a continuous probability density function and then derive it for the Gaussian distribution.

The median is referred to as the **unbiased estimator**, as it is the statistic where the distribution lies 50% on either side. To obtain this statistic, we need to evaluate the cumulative density function to 0.5. Therefore, the median is defined as the state, x_{med} such that

$$\int_a^{x_{med}} f(x)\, dx = 0.5. \tag{3.130}$$

Returning to the Gaussian distribution, we know that it is not possible to integrate analytically the cumulative density function for this distribution. However, many textbooks contain the tables for the cumulative density function for the Gaussian distribution. Therefore, you can check and see that the cumulative density function is equal to 0.5 when $x = \mu$. Therefore, the median is equal to the mean for the Gaussian distribution.

3.7.5 Mode of the Univariate Gaussian Distribution

The final descriptive statistic that we introduce is the mode. The mode is the **most likely** or **maximum likelihood state**. It is the value of the random variable with the highest probability. The mode is found by differentiating the probability density function and finding the value for the random variable where the derivative is equal to zero. This is defined mathematically as

$$x_{mode} \quad s.t. \quad \left.\frac{df}{dx}\right|_{x=x_{mode}} = 0.$$

Therefore, to find the mode of the Gaussian distribution, we require the derivative of

$$N\left(\mu,\sigma^2\right)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}.$$

The derivation for the mode of the Gaussian distribution is as follows:

$$\begin{aligned}\frac{dN}{dx}&=-\frac{1}{\sqrt{2\pi}\sigma}\frac{(x-\mu)}{\sigma^2}\exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}=0,\\&=-(x-\mu)\exp\left\{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right\}=0,\\&=-x+\mu=0\Rightarrow x=\mu,\end{aligned}$$

which is again equal to the mean.

Therefore, all three descriptive statistics for the Gaussian distribution are equal. The next distribution that we consider is the **lognormal distribution**.

3.8 Lognormal Distribution

The lognormal is considered the first step away from the Gaussian distribution and is the only distribution that has an invertible transformation to the Gaussian distribution. The definition for the univariate lognormal PDF is

$$LN\left(\mu,\sigma^2\right)\equiv\frac{1}{\sqrt{2\pi}\sigma}\frac{1}{x}\exp\left\{-\frac{1}{2}\frac{(\ln x-\mu)^2}{\sigma^2}\right\},\quad x\in(0,\infty),\tag{3.131}$$

where one of the important features of the lognormal distribution is

$$\mu\equiv\mathbb{E}\left[\ln x\right],\qquad\sigma^2\equiv\mathbb{E}\left[(\ln x)^2\right]-\left(\mathbb{E}\left[\ln x\right]\right)^2.$$

Therefore, the parameters of the distribution are in terms of $\log x$ **not** x. A second important feature of the lognormal distribution is that it is defined for $x\in(0,\infty)$, not $x\in(-\infty,\infty)$ or $x\in[0,\infty)$, which implies the distribution is defined for positive definite random variables, i.e., greater than but never equal to zero. A random variable that can obtain the value zero but not go negative is referred to as a semi-positive definite random variable.

3.8.1 Moments of the Lognormal Distribution

Now we derive the first moment of the lognormal distribution, which comes from

$$\mathbb{E}\left[X\right]=\frac{1}{\sqrt{2\pi}\sigma}\int_0^\infty\exp\left\{-\frac{1}{2}\frac{(\ln x-\mu)^2}{\sigma^2}\right\}dx.\tag{3.132}$$

To proceed further, the change of variable $y = \ln x \Rightarrow x = \exp\{y\}$ and $dy = x^{-1}dx \Rightarrow dx = \exp\{y\}\,dy$ is introduced. The associated limits of integration for the new variable are $x = 0 \Rightarrow y = -\infty$ and $x = \infty \Rightarrow y = \infty$. This then makes (3.132) in terms of y as

$$\mathbb{E}[X] = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}\exp\left\{-\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}\right\}\exp\{y\}\,dy.$$

The next step is to combine the exponentials' exponent, which results in

$$\frac{-y^2 + 2\left(\mu+\sigma^2\right)y - \mu^2}{2\sigma^2}.$$

As with the Gaussian distribution, we complete the square on the expression above, which yields

$$-\frac{1}{2}\frac{\left(y-\left(\mu+\sigma^2\right)\right)^2}{\sigma^2} + \frac{2\mu\sigma^2+\sigma^4}{2\sigma^2}.$$

This allows (3.132) to be rewritten as

$$E[x] = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}\exp\left\{-\frac{1}{2}\frac{\left(y-\left(\mu+\sigma^2\right)\right)^2}{\sigma^2} + \left(\mu+\frac{\sigma^2}{2}\right)\right\}.$$

Now, using the property that the sum of exponentials is the equivalent to the product of two exponentials, and noticing that the second exponential is not a function of y which implies that it can be taken out of the integral, leads to

$$\begin{aligned}\mathbb{E}[x] &= \exp\left\{\left(\mu+\frac{\sigma^2}{2}\right)\right\}\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}\exp\left\{-\frac{1}{2}\frac{\left(y-\left(\mu+\sigma^2\right)\right)^2}{\sigma^2}\right\},\\ &= \exp\left\{\left(\mu+\frac{\sigma^2}{2}\right)\right\}. \end{aligned} \tag{3.133}$$

An important feature of the lognormal mean is that it is unbounded with respect to the variance of $\log x$. Therefore, as $\sigma^2(\log x) \to \infty \Rightarrow \mu(x) \to \infty$.

Now we move on to the second non-central moment and the variance. The starting point is $\mathbb{E}\left[X^2\right]$, which is

$$\mathbb{E}\left[X^2\right] = \frac{1}{\sqrt{2\pi}\sigma}\int_0^{\infty}\frac{x^2}{x}\exp\left\{-\frac{1}{2}\frac{(\ln x-\mu)^2}{\sigma^2}\right\}dx. \tag{3.134}$$

Introducing the change of variable used in the derivation of the lognormal mean results in

$$\mathbb{E}\left[X^2\right] = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}\exp\left\{-\frac{1}{2}\frac{(y-\mu)^2}{\sigma^2}\right\}\exp\{2y\}\,dy.$$

After combining the exponents, the resulting expression that we have to complete the square on is

$$\frac{-y^2 + 2\mu y + 4\sigma^2 y - \mu^2}{2\sigma^2},$$

which can easily be shown to be

$$-\frac{1}{2\sigma^2}\left(y-\left(\mu+2\sigma^2\right)\right)^2+\frac{1}{2\sigma^2}\left(4\mu\sigma^2+4\sigma^4\right).$$

Recognizing that the second term above is constant with respect to y, then it can easily be shown that the lognormal non-central second moment is

$$\mathbb{E}\left[X^2\right]=\exp\left\{2\mu+2\sigma^2\right\}. \tag{3.135}$$

This then leads to the lognormal variance being

$$\begin{aligned} VAR(X) &= \mathbb{E}\left[X^2\right]-(\mathbb{E}[X])^2, \\ &= \exp\left\{2\mu+2\sigma^2\right\}-\left(\exp\left\{\mu+\frac{\sigma^2}{2}\right\}\right)^2, \\ &= \exp\left\{2\mu+\sigma^2\right\}\left(\exp\left\{\sigma^2\right\}-1\right). \end{aligned} \tag{3.136}$$

We now consider the effects that the two parameters μ $(\log x)$ and σ^2 $(\log x)$ have on the appearance of the shape of the lognormal distribution. In Fig. 3.8 we have fixed the variance of $\log x$ and allowed the

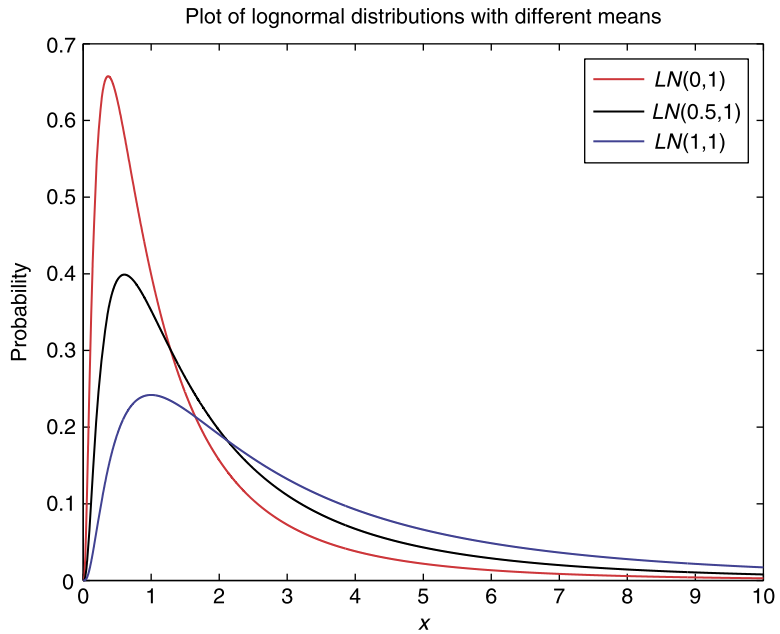


FIGURE 3.8

Plots of lognormal distribution with constant variance but changing means.

mean to be 0, 0.5, and 1. We see that the appearance of the lognormal distribution is skewed to the left, but that the three distributions appear to have similar shapes.

However, in Fig. 3.9 we have plotted three different lognormal distribution where we have kept the mean constant at $\mu(\log x) = 0$ and allowed the variance to have the values 0.0625, 1, and 4. It is clear that for this situation the shape of the distribution is affected by changing the variance parameter. We can see that the state with the highest probability decreases toward zero as the variance increases. We also see that when the variance is small, the distribution almost appears to be symmetric, and hence almost Gaussian, but there is still a slight left skewness. If we consider the two plots together, then it would appear that the skewness is not affected as much by the mean changing as it is by the variance. To verify why this might be, we now derive the skewness and the kurtosis coefficients for the lognormal distribution to see what their relationships are to $\mu(\log x)$ and $\sigma(\log x)$.

The derivation of the skewness and kurtosis is not as straightforward as for the Gaussian distribution. While some information outlets say there is a MGF for the lognormal distribution, this is not true. It is not possible to analytically evaluate the integral to derive the MGF for the lognormal distribution; in addition, the uniqueness theorem prevents the lognormal distribution from having a MGF. In [173] it is shown that the PDF given by

$$P(x) = \frac{1}{\left(\sigma(x-a)\sqrt{2\pi}\right)} \exp\left\{-\frac{1}{2}\left(\frac{\ln(x-a)-\mu}{\sigma}\right)^2\right\}\left(1+\epsilon\sin\left(\frac{2\pi k(\ln(x-a)-\mu)}{\sigma^2}\right)\right), \tag{3.137}$$

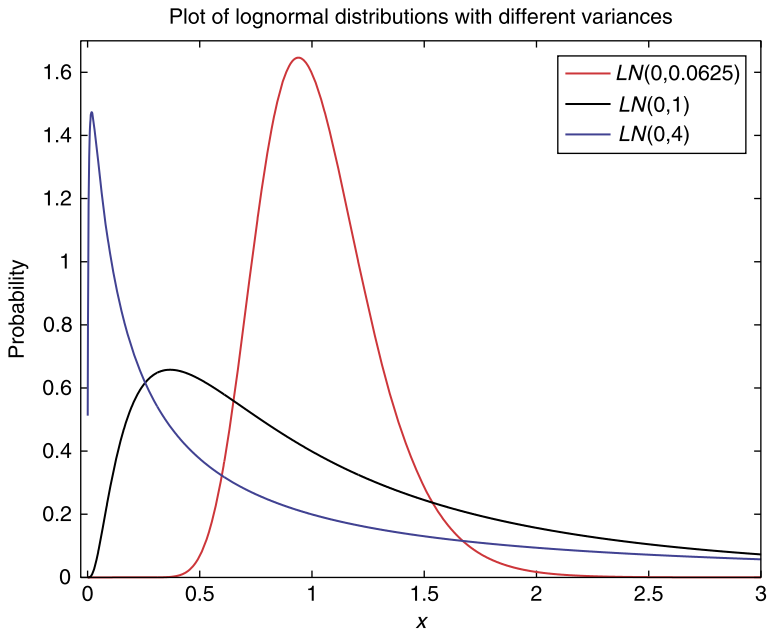


FIGURE 3.9

Plots of lognormal distribution with constant mean but changing variances.

where $0 < \epsilon < 1$ and k is a positive integer, has the same moments to infinity as the lognormal distribution. Therefore, a MGF for a lognormal distribution would also be the MGF for the distribution above, but the uniqueness theorem states that this cannot happen. Therefore, there cannot exist a MGF for the lognormal distribution.

However, it is often stated that by using the property that the log of a lognormal random variable is a Gaussian random variable, we can simply substitute this property into the Gaussian distribution's MGF and obtain an expression for the lognormal distribution. This is not true; however, there is a general form for the non-central moments given by

$$\mathbb{E}\left(X^n\right) = \exp\left\{n\mu + \frac{n^2\sigma^2}{2}\right\}. \tag{3.138}$$

Using the expression in (3.138), we can obtain the expression for the third non-central moment of the lognormal distribution as

$$\mathbb{E}\left[X^3\right] = \exp\left\{3\mu + \frac{9\sigma^2}{2}\right\}. \tag{3.139}$$

To obtain the expression for the skewness, we need to evaluate

$$\beta_1 = \frac{\mathbb{E}\left[X^3\right] - 3\mathbb{E}[X]\,\mathbb{E}\left[X^2\right] + 2\left(\mathbb{E}[X]\right)^3}{(Var[X])^{\frac{3}{2}}},$$

which for the lognormal distribution is

$$\beta_1 = \frac{\exp\left\{3\mu + \frac{9\sigma^2}{2}\right\} - 3\exp\left\{3\mu + \frac{5\sigma^2}{2}\right\} + 2\exp\left\{3\mu + \frac{3\sigma^2}{2}\right\}}{\left(\exp\left\{2\mu + \sigma^2\right\}\left(\exp\left\{\sigma^2\right\} - 1\right)\right)^{\frac{3}{2}}}. \tag{3.140}$$

The first thing to notice about (3.140) is that the $\exp\{3\mu\}$ terms cancel each other. This proves the statement that the skewness of the lognormal distribution is independent of the mean of the log of the lognormal random variable. Another feature to notice is that it is possible to factorize out $\exp\left\{\frac{3\sigma^2}{2}\right\}$ from the numerator and cancel the same term in the denominator.

To finish deriving the skewness coefficient of the lognormal distribution, we introduce a change of variable to make the derivation clearer to follow. Therefore, let $\Omega = \exp\left\{\sigma^2\right\}$ and rewrite (3.140) after the cancelations as

$$\beta_1 = \frac{\Omega^3 - 3\Omega + 2}{(\Omega - 1)\sqrt{(\Omega - 1)}}.$$

After some polynomial long division, it is possible to write the skewness for the lognormal distribution as

$$\beta_1 = \frac{(\Omega - 1)(\Omega - 1)(\Omega + 2)}{(\Omega - 1)\sqrt{(\Omega - 1)}} = \sqrt{(\Omega - 1)}(\Omega + 2) = \sqrt{\left(\exp\left\{\sigma^2\right\} - 1\right)}\left(\exp\left\{\sigma^2\right\} + 2\right). \tag{3.141}$$

We now consider the kurtosis of the lognormal distribution. Given the expression for the nth non-central moment, (3.138), we know that the fourth non-central moment for the lognormal distribution is

$$\mathbb{E}\left[X^4\right] = \exp\left\{4\mu + 8\sigma^2\right\}.$$

Evaluating the expression for the kurtosis in terms of the different non-central moments of the lognormal distribution and canceling with terms in the denominator, which is left as an exercise, the kurtosis for a lognormal distribution can be shown to be

$$\beta_2 = \exp\left\{4\sigma^2\right\} + 2\exp\left\{3\sigma^2\right\} + 3\exp\left\{\sigma^2\right\} - 3. \tag{3.142}$$

Exercise 3.41. *Verify that the kurtosis of the univariate lognormal distribution is* (3.142).

3.8.2 Geometric Behavior of the Lognormal

An important property of the lognormal distribution is that it is of the geometric type of distributions. Thus we have that the product of two independent lognormally distributed random variables, $X_1 \sim LN\left(\mu_1, \sigma_1^2\right)$ and $X_2 \sim LN\left(\mu_2, \sigma_2^2\right)$, denoted by $X_1 X_2$, is also a lognormally distributed random variable such that $X_1 X_2 \sim LN\left(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2\right)$. This condition can be generalized to N lognormal random variables as

$$\prod_{i=1}^{N} X_i \sim LN\left(\sum_{i=1}^{N}\mu_i, \sum_{i=1}^{N}\sigma_i^2\right), \quad X_i \sim LN\left(\mu_i, \sigma_i^2\right), \quad i = 1, 2, \ldots, N. \tag{3.143}$$

The same property is true for the quotient of two independently distributed lognormal random variables, i.e., if $X_1 \sim LN\left(\mu_1, \sigma_1^2\right)$ and $X_2 \sim LN\left(\mu_2, \sigma_2^2\right)$, then $\frac{X_1}{X_2} \sim LN\left(\mu_1 - \mu_2, \sigma_1^2 + \sigma_2^2\right)$. The quotient property will become important when we consider how to define the errors for a lognormal distribution variational-based data assimilation system later.

3.8.3 Median of the Univariate Lognormal Distribution

The derivation of the median for the lognormal distribution uses the established property that the logarithm of a lognormal random variable is a Gaussian random variable. Therefore, if we have $X \sim LN\left(\mu, \sigma^2\right)$ then $\log X \sim N\left(\mu, \sigma^2\right)$. We have already shown that the median of a Gaussian distributed random variable is equal to the mean, μ. Therefore we have that the median of $\log X$, $(\log X)_{med} = \mu$. Thus, through the preservation of percentiles when transforming between distribution, and recalling that the median is the 50% percentile, by inverting the logarithm we have that the median of the lognormally distributed random variable as $x_{med} = \exp\{\mu\}$.

We now have expressions for two of the three descriptive statistics for the lognormal distribution, and we can see that there is a difference between the Gaussian and lognormal distributions. For the lognormal distribution, the two descriptive statistics so far are not equal; in fact we have that

$$X_{med} < X_{mean} \equiv \exp\{\mu\} < \exp\left\{\mu + \frac{\sigma^2}{2}\right\}.$$

The expression above can only be equal in the case of $X = c$, which is when $\sigma^2 = 0$.

3.8.4 Mode of the Lognormal Distribution

The mode, as explained earlier, is the maximum likelihood state, i.e., the state with the highest probability of occurring. This is when the derivative of the PDF is equal to zero. Therefore, applying this to the lognormal PDF we have

$$X_{mode}\ s.t.\ \frac{d}{dx}\left(\frac{1}{\sqrt{2\pi}\sigma}\frac{1}{x}\exp\left\{-\frac{1}{2}\frac{(\ln x-\mu)^2}{\sigma^2}\right\}\right)\Bigg|_{X_{mode}}=0. \tag{3.144}$$

As (3.144) is a product of functions of x, we apply the product rule, which gives

$$-\frac{1}{x^2}\left(1+\frac{\ln x-\mu}{\sigma^2}\right)\exp\left\{-\frac{1}{2}\frac{(\ln x-\mu)^2}{\sigma^2}\right\}=0. \tag{3.145}$$

The only way (3.145) can be zero is if the expression in the bracket is equal to zero. Therefore, through some rearrangement we have

$$X_{mode}=\exp\left\{\mu-\sigma^2\right\}. \tag{3.146}$$

Given the expression for the mode of the lognormal distribution in (3.146), we now have the inequality linking the three descriptive statistics for the lognormal distribution as

$$\begin{aligned} X_{mode} &< X_{med} < X_{mean}, \\ \exp\left\{\mu-\sigma^2\right\} &< \exp\{\mu\} < \exp\left\{\mu+\frac{\sigma^2}{2}\right\}. \end{aligned} \tag{3.147}$$

The property in (3.147) will become quite important when deciding how to start the derivation of variational data assimilation system for lognormally distributed errors.

To illustrate how the three descriptive statistics are affected by the values of μ and σ, we have plotted different lognormal distributions in Fig. 3.10. When σ^2 is small, there are a couple of features that we notice. The first feature is that the lognormal distribution looks approximately like a Gaussian distribution; this is consistent with the skewness of the lognormal distribution tending to zero, as σ^2 tends toward zero. The second feature about the lognormal distribution for small variances is that the mean (green line in the plots) and the mode (blue line in the plots) are converging toward the median (red line in the plots), with the mean closer to the median above than the mode is below.

We have plotted three other lognormal distributions with their three descriptive statistics marked on each one using the same colors as mentioned above. The interesting feature about the other three plots in (3.10) is that as the variance increases, the median does not change. However, we see that the mean is moving further away into the tail of the distribution; while the mode is decaying toward zero, it is not as fast as the movement away that the mean has relative to the median. We shall re-examine this feature in Chapter 21, as it has significant consequences involving transforming lognormal random variables into Gaussian-distributed random variables.

The lognormal distribution is a very important distribution after the Gaussian distribution for geophysical processes. When writing the first edition the lognormal distribution was the only other distribution besides a Gaussian that had a variational data assimilation defined for it [129,132,135,137].

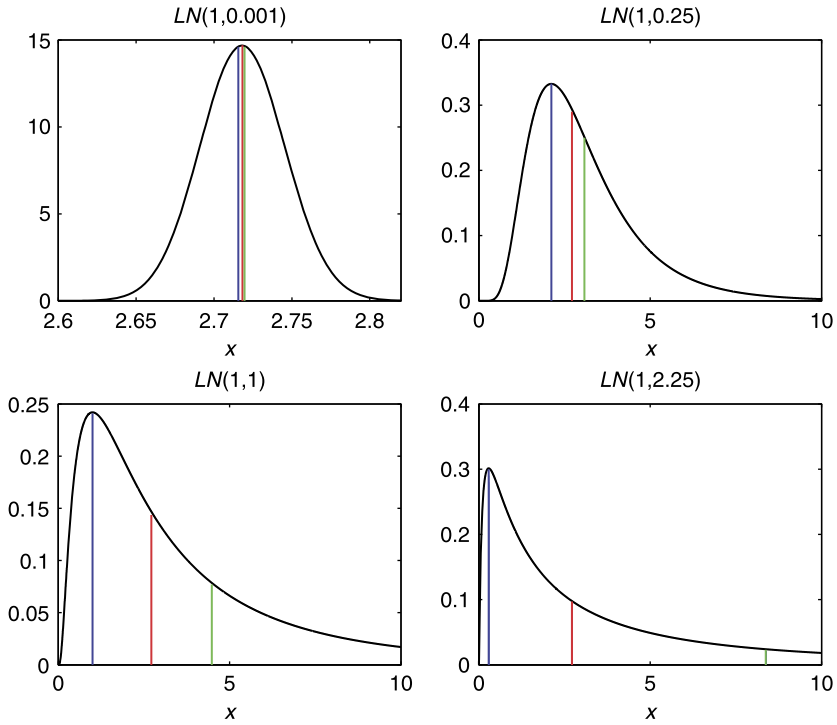


FIGURE 3.10

Plots of lognormal distributions with increasing variance, to highlight how the three descriptive statistics diverge from each other, where blue is the mode, red is the median, and green is the mean.

It has been shown that mixing ratios at certain levels of the atmosphere follows a lognormal distribution: by three-month groupings and by year [221]. In [129] it is shown that cloud liquid water taken over a four-year period at the Atmospheric Radiation Measurement (ARM) South Great Plains (SGP) site in Oklahoma between 2001 and 2004 when a boundary layer cloud was present followed different distributions depending on the season, but was shown to be lognormally distributed in when the distribution was condition on the winter and spring season. The summer season had a clear normal distribution signal, yet in the fall season we can see a transition between the two distributions resulting in a bi-modal distribution. In [409] the motivation for the authors' work was to devise an incremental lognormal-based 4D VAR system, which shall be explained in Chapter 21, to forecast where a biological substance would be on the ocean surface, where the Gaussian-based incremental version of variational data assimilation would periodically create too big a negative increment, so that the analysis state would become unphysical.

Perhaps the most important property of the lognormal distribution is its equivalency to the additive central limit theorem. The equivalency results in what is called the multiplicative central limit theorem.

Theorem 3.42. ***Multiplicative Central Limit Theorem****: If* $\{Z_i\}$ *is a sequence of positive, independent, and identically distributed random variables such that*

$$E[\ln Z_i] = \mu < \infty,$$
$$VAR[\ln Z_i] = \sigma^2 < \infty,$$

then the product $\prod_{i=1}^{n} Z_i$ *is asymptotically distributed as a lognormal random variable with* $LN(n\mu, n\sigma^2)$.

Here we now introduce a new distribution that was not in the first edition. Through work at the Cooperative Institute for Research in the Atmosphere (CIRA) at Colorado State University (CSU), a new distribution was discovered to be present when considering the z component of the Lorenz 63 model, [270], [158] and as such a new version of variational data assimilation was developed. This new distribution has also been discovered to describe water vapor. We refer to this distribution as the **reverse lognormal** and we introduce this distribution next.

3.9 Reverse Lognormal Distribution

As just mentioned recent research that we have been part of at CIRA/CSU has involved using machine learning to detect distribution changes in the Lorenz 1963 model, [270], in [158,159] where in the latter we used the machine learning to decide when to minimize a Gaussian or a lognormal cost function, we shall explain this when we arrive at the relevant chapters. What came of out both [158] and [159] was the detection of a right skewed distribution that was neither Gaussian, nor lognormal. A similar distribution had also been found in [220]. It became apparent that this distribution was in fact a **reverse lognormal**.

Until recently, when doing the research for this second edition, the reverse lognormal was not a distribution we were familiar with. It has, however, been known about for quite a while. In [142] the reverse lognormal distribution is presented as part of the three parameter lognormal distribution.

The three parameter lognormal distribution is still defined by its mean μ, variance σ^2, but now there is a third term referred to as the *threshold* parameter, that allows the distribution to describe the situation where the variable has a nonzero lower bound. The threshold simply acts to translate the PDF along the x-axis. In [142] the lognormal distribution is denoted by Λ, and the three parameter version is identified by $\Lambda(x \mid \mu, \sigma, t)$. However, Foster et al. in [142] refer to the parameters as M for μ but call it the median, and s for σ, but that this is the geometric standard deviation. This is referring to the property that these parameters are of $\ln x$ and not x. Therefore, the three parameter lognormal distribution is given by

$$\Lambda\left(x \mid \mu, \sigma^2, t\right) \equiv \frac{1}{(x-t)\,\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{(x-t)-\mu}{\sigma}\right)^2\right\} \qquad t < x < \infty. \tag{3.148}$$

The reverse lognormal is obtained through defining the new variate $x' = t - x$, which is distributed according to the two parameter lognormal distribution that we showed in the last section. However, we

can also define the reverse lognormal distribution, $R\Lambda$, as

$$R\Lambda\left(x|\mu,\sigma^2,t\right) \equiv \frac{1}{(t-x)\,\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\left(\frac{(t-x)-\mu}{\sigma}\right)^2\right\} \quad = \infty < x < t. \tag{3.149}$$

In Fig. 3.11 we have recreated Figure 1 from [142] that shows the lognormal (LN), $\Lambda\left(x|0,0.7^2\right)$, the shifted lognormal (SLN), $\Lambda\left(x|0,0.7^2,2\right)$, and the reverse lognormal (RLN), $\Lambda\left(5-x|0,0.7^2\right)$, where we can see the structure of the three distributions.

3.9.1 Mean of the Reverse Lognormal Distribution

As the reverse lognormal is part of the three-parameter lognormal family, there is no moment generating function, so we will use the integral approach to find the non-central moments. Therefore, the expectation of x for a reverse lognormal distribution is given by

$$\mathbb{E}[X] = \int_{-\infty}^{t} \frac{x}{(t-x)\,\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\left(\frac{\ln(t-x)-\mu}{\sigma}\right)^2\right\} dx. \tag{3.150}$$

We again introduce a change of variable so that the new random variable follows a Gaussian distribution. This time the change of variable is $y \equiv \ln(t-x) \Rightarrow x \equiv t - e^y$. Looking at the limits of

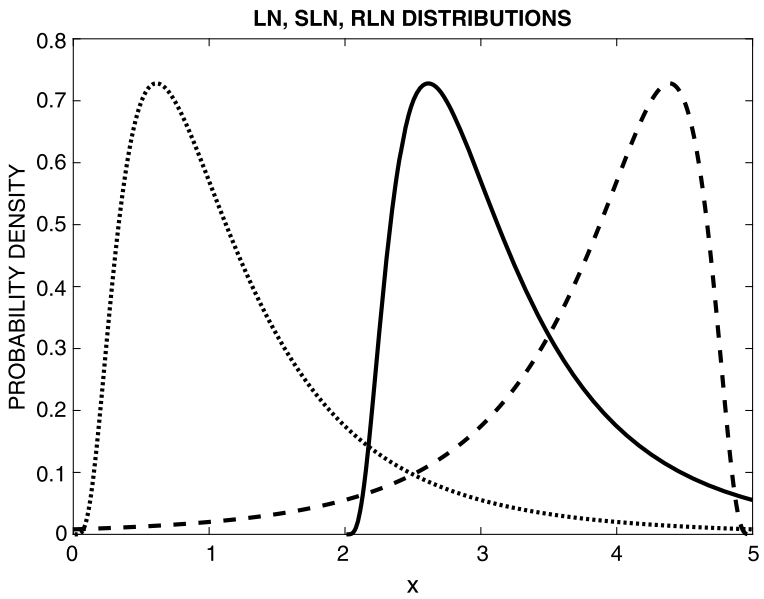


FIGURE 3.11

Recreation of Fig. 1 from [142] that show three different lognormal distributions from the three parameter family, where the dot plot is the lognormal (LN), $\Lambda\left(x|0,0.7^2\right)$, the solid line is the shifted lognormal (SLN), $\Lambda\left(x|0,0.7^2,2\right)$, and the dashed line is the reverse lognormal (RLN), $\Lambda\left(5-x|0,0.7^2\right)$.

integration, when $x = t$ then $y = -\infty$, and when $x = -\infty$, $y = \infty$, which implies that the limits are round the wrong way, but we have one more steps. The step is $dx = -e^y$, and it is the minus sign here that allows us to swap the limits around. Given this information we now have

$$\mathbb{E}[Y] = \int_{-\infty}^{\infty} \frac{(t - \exp\{y\})}{(t - t + \exp\{y\})\,\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{y-\mu}{\sigma}\right)^2\right\} \exp\{y\}\,dy. \tag{3.151}$$

Now we notice that the ts in the denominator of the scaling factor cancel, as does the exponential in the denominator with the exponential multiplying the integrand, we see that we have the term $t - \exp\{y\}$ multiplying the Gaussian CDF. Therefore the term multiplying t integrates to 1, where the second term is the same term we obtained during the derivation of the lognormal mean. This implies that the mean of the reverse lognormal is

$$x_{mean} = t - \exp\left\{\mu + \frac{\sigma^2}{2}\right\}. \tag{3.152}$$

3.9.2 Variance of the Reverse Lognormal Distribution

To derive the variance we require the second order non-central moment of the reverse lognormal which starts from

$$\mathbb{E}\left[X^2\right] = \int_{-\infty}^{t} \frac{x^2}{(t-x)\,\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{\ln(t-x)-\mu}{\sigma}\right)^2\right\} dx. \tag{3.153}$$

We introduce the change of variable $y \equiv \ln(t - x)$, so that $x = t - \exp\{y\}$. In this case we have that $x^2 \equiv t^2 - 2t\exp\{y\} + \exp\{2y\}$. While at first this may appear daunting, it is not due to the fact that we have seen these expression before. As t^2 is not a function of y, the integral multiplying it is the CDF of a Gaussian which equals 1 when evaluated over all of its values. The terms $-2t\exp\{y\}$ we have just seen is the expression for the lognormal mean, so this gives us $-2t\exp\left\{\mu + \frac{\sigma^2}{2}\right\}$. The final term is the expression for the lognormal second order non-central moment; $\exp\left\{2\mu + 2\sigma^2\right\}$. Recalling the equation for the variance, we obtain

$$\begin{aligned}
VAR[X] &= \mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2, \\
&= t^2 - 2t\exp\left\{\mu + \frac{\sigma^2}{2}\right\} + \exp\left\{2\mu + 2\sigma^2\right\} - \left(t - \exp\left\{\mu + \frac{\sigma^2}{2}\right\}\right)^2, \\
&= t^2 - 2t\exp\left\{\mu + \frac{\sigma^2}{2}\right\} + \exp\left\{2\mu + 2\sigma^2\right\} - t^2 + 2t\exp\left\{\mu + \frac{\sigma^2}{2}\right\} - \exp\{2\mu + \sigma\}, \\
&= \exp\left\{2\mu + 2\sigma^2\right\} - \exp\left\{2\mu + \sigma^2\right\}, \\
&= \exp\left\{2\mu + \sigma^2\right\}\left(\exp\left\{\sigma^2\right\} - 1\right).
\end{aligned} \tag{3.154}$$

Therefore, the variance of the reverse lognormal is the same as the lognormal distribution. This makes sense as this is a measure of spread and as such the reflection would have the same spread.

3.9.3 Skewness of the Reverse Lognormal Distribution

We now move on to the skewness, where we know that we need to evaluate $\mathbb{E}\left[X^3\right]$. We present this derivation in a slightly different way as it is clear from the last two derivations, that the non-central moments can be expressed in terms of the expectation of a lognormal random variable. Through introducing the same change of variable as earlier, we have $x^3 \equiv (t - e^y)^3 = t^3 - 3t^2e^y + 3te^y + e^{3y}$. When this is substituted into the integral from earlier we can write $\mathbb{E}\left[X^3\right]$ in terms of the lower order non-central moments of the reverse lognormal distribution, and then in term of the lognormal non-central moments. Therefore for the first term in the skewness derivation we have

$$\begin{aligned}
\mathbb{E}\left[X^3\right] &= t^3 - 3t^2\mathbb{E}[X] + 3t\mathbb{E}\left[X^2\right] - \mathbb{E}\left[Y^3\right], \\
&= t^3 - 3t^2(t - \mathbb{E}[Y]) + 3t\left(t^2 - 2t\mathbb{E}[Y] + \mathbb{E}\left[Y^2\right]\right) - \mathbb{E}\left[Y^3\right], \\
&= t^3 - 3t^3 + 3t^2\mathbb{E}[Y] + 3t^3 - 6t\mathbb{E}[Y] + 3t\mathbb{E}\left[Y^2\right] - \mathbb{E}\left[Y^3\right], \\
&= t^3 - 3t^2\mathbb{E}[Y] + 3t\mathbb{E}\left[Y^2\right] - \mathbb{E}\left[Y^3\right], \qquad (3.155)
\end{aligned}$$

where Y is a lognormal random variable. Recalling that the numerator of the definition of skewness is given by $\mathbb{E}\left[X^3\right] - 3\mathbb{E}[X]\mathbb{E}\left[X^2\right] + 2\mathbb{E}[X]^3$, we now consider the second term;

$$\begin{aligned}
-3\mathbb{E}[X]\mathbb{E}\left[X^2\right] &= -3(t - \mathbb{E}[Y])\left(t^2 - 2t\mathbb{E}\left[Y^2\right] + \mathbb{E}\left[Y^2\right]\right), \\
&= -3t^3 + 6t^2\mathbb{E}[Y] - 3t\mathbb{E}\left[Y^2\right] + 3t^2\mathbb{E}[Y] - 6t\mathbb{E}[Y]^2 + 3\mathbb{E}[Y]\mathbb{E}\left[Y^2\right], \\
&= -3t^3 + 9t^2\mathbb{E}[Y] - 3t\mathbb{E}\left[Y^2\right] - 6t\mathbb{E}[Y]^2 + 3\mathbb{E}[Y]\mathbb{E}\left[Y^2\right]. \qquad (3.156)
\end{aligned}$$

Consider the final term in the numerator of the skewness coefficient we have

$$\begin{aligned}
+2\mathbb{E}\left[X^3\right] &= 2(t - \mathbb{E}[Y])^3, \\
&= 2t^3 - 6t^2\mathbb{E}[Y] + 6t\mathbb{E}[Y]^2 - 2\mathbb{E}[Y]^3. \qquad (3.157)
\end{aligned}$$

Combining the terms in (3.155)–(3.157) results in all of the terms that contain powers of t canceling, where the remaining terms are: $-\mathbb{E}[Y] + 3\mathbb{E}[Y]\mathbb{E}\left[Y^2\right] - 2\mathbb{E}[Y]^3$. When these terms are combined with the denominator of β_1, which is in terms of the variance, that we know is the same as the lognormal distribution variance, then the skewness of the reverse lognormal distribution is the negative of the skewness of its lognormal counterpart.

3.9.4 Kurtosis of the Reverse Lognormal Distribution

As with the other moments, we introduce the change of variable to form $\mathbb{E}\left[X^4\right]$, which gives us on the numerator $(t - e^y) = t^4 - 4t^3e^y + 6t^2e^{2t} - 4te^{3y} + e^{4y}$. This then implies $\mathbb{E}\left[X^4\right]$ is equal to

$$\begin{aligned}
\mathbb{E}\left[X^4\right] = t^4 &- 4t^3(t - \mathbb{E}[Y]) + 6t^2\left(t^2 - 2t\mathbb{E}[Y] + \mathbb{E}\left[X^2\right]\right) \\
&- 4t\left(t^3 - t^2\mathbb{E}[Y] + 3t\mathbb{E}[Y]^2 - \mathbb{E}\left[Y^3\right]\right) + \mathbb{E}\left[Y^4\right].
\end{aligned}$$

The derivation of kurtosis is left as an exercise.

Exercise 3.43. *Derived the kurtosis coefficient of the reverse lognormal distribution.*

3.9.5 Median of the Reverse Lognormal Distribution

As we saw with the lognormal distribution, that through the preservation of percentiles, which the median is the 50% percentiles. Therefore, introducing the change of variable $Y = t - \exp\{x\}$ we know this is a Gaussian random variable and has a median of μ. Now through the preservation of percentiles we have $Y_{median} = \mu$. Thus inverting the transformation we that the median of the reverse lognormal distribution is given by

$$x_{med} = t - \exp\{\mu\}. \tag{3.158}$$

Note: That this is similar in appearance to the mean in that it is the threshold parameter minus the equivalent descriptive statistic of the lognormal distribution.

3.9.6 Mode of the Reverse Lognormal Distribution

As we know the mode is the most likely state and so we have to differentiate the reverse lognormal distribution and set the gradient equal to zero to find this state, we do as follows

$$\begin{aligned}
\frac{d\Lambda\left(x|\mu,\sigma^2,t\right)}{dx} &= \frac{d}{dx}\left(\frac{1}{(t-x)\sigma\sqrt{2\pi}}\exp\left\{-\frac{1}{2}\left(\frac{\ln(t-x)-\mu}{\sigma}\right)^2\right\}\right) = 0, \\
&= \frac{1}{(t-x)^2\sigma\sqrt{2\pi}}\left(1-\frac{\ln(t-x)-\mu}{\sigma^2}\right)\exp\left\{-\frac{1}{2}\left(\frac{\ln(t-x)-\mu}{\sigma}\right)^2\right\} = 0, \\
&\Rightarrow 1-\frac{\ln(t-x)-\mu}{\sigma^2} = 0, \\
&\Rightarrow x_{mode} = t - \exp\left\{\mu-\sigma^2\right\}.
\end{aligned} \tag{3.159}$$

Thus the mode of the reverse lognormal distribution is again the threshold parameter minus the equivalent descriptive statistic of the lognormal distribution.

Therefore to summarize we have that the mean is less than the median, which is less than the mode for the reverse lognormal distribution:

$$t - \exp\left\{\mu+\frac{\sigma^2}{2}\right\} \leq t - \exp\{\mu\} \leq t - \exp\left\{\mu-\sigma^2\right\}. \tag{3.160}$$

We now move on to consider some of the more nonlinear defined PDFs.

3.10 Exponential Distribution

The exponential distribution is another probability distribution for positive definite random variables, i.e., $X > 0$. The PDF for an exponentially distributed random variable, X, is defined as

$$p(x) = \frac{1}{\sigma}\exp\left\{-\frac{(x-\mu)}{\sigma}\right\}, \quad x > 0, \mu > 0. \tag{3.161}$$

As with the Gaussian distribution, there is an equivalent *standardized* version of the exponential distribution when $\mu = 0$ and $\sigma = 1$, who's PDF is given by

$$p(x) = \exp\{-x\}, \quad x > 0.$$

The MGF of the exponential distribution, as defined in (3.161), can be shown to be

$$M_e(t) \equiv \frac{1}{(1-\sigma t)} \exp\{t\mu\}. \tag{3.162}$$

Exercise 3.44. *Derive the moment-generating function of the exponential distribution given in* (3.162).

Exercise 3.45. *Verify that the mean, variance, and the non-central third and fourth moments are*

$$\mu_1 = \sigma, \quad VAR(X) = \sigma^2, \quad \mu_3 = 2\sigma^3, \quad \mu_4 = 9\sigma^4. \tag{3.163}$$

It is possible to show that the mode of the exponential distribution is $x_{mode} = \mu$ and the median is $x_{med} = \sigma \ln 2$.

To help visualize the exponential distribution, we have plotted three different exponential distributions for the cases of $\sigma^{-1} = 0.5$, $\sigma^{-1} = 1$, and $\sigma^{-1} = 1.5$ in Fig. 3.12. We can see that this distribution has quite a stark difference in its shape compared to the Gaussian and the lognormal distribution, but is a left-skewed distribution.

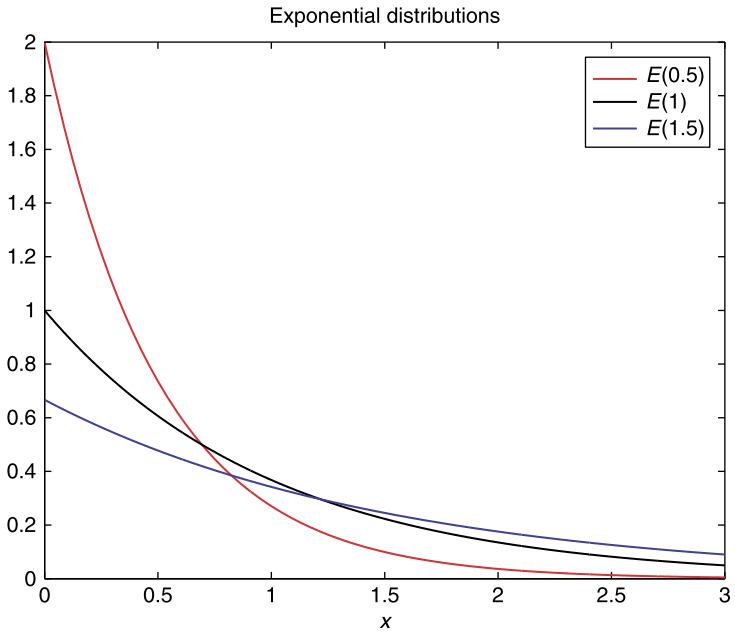


FIGURE 3.12

Plots of different versions of the exponential distribution.

The exponential distribution occurs naturally when describing the lengths of inter-arrival times in a homogeneous Poisson process. The exponential distribution is sometimes referred to as the continuous counterpart to the discrete geometric distribution, which has already been shown to describe the number of Bernoulli trials necessary for a discrete process to change state. Therefore, the exponential distribution represents the time for a continuous process to change state.

The exponential distribution has been used to analyze extreme values of monthly and annual maximum rainfall and river discharge volumes.

3.11 Gamma Distribution

Before introducing the gamma distribution, we require the **gamma function**. This function, denoted $\Gamma(\alpha)$, is defined as

$$\Gamma(\alpha) \equiv \int_0^\infty x^{\alpha-1} \exp\{-x\}\, dx. \tag{3.164}$$

To solve the integral in (3.164), we deploy integration by parts again, where we have

$$\begin{aligned} u(x) &= x^{\alpha-1}, & v'(x) &= \exp\{-x\}, \\ u'(x) &= (\alpha-1)\, x^{\alpha-2}, & v(x) &= -\exp\{-x\}, \end{aligned}$$

which results in

$$\Gamma(\alpha) = \left[-x^{\alpha-1} \exp\{-x\}\right]_0^\infty + (\alpha-1) \int_0^\infty x^{\alpha-2} \exp\{-x\}\, dx.$$

The first part of the expression above can easily be shown to be equal to zero, and therefore we have

$$\Gamma(\alpha) = (\alpha-1) \int_0^\infty x^{\alpha-2} \exp\{-x\}\, dx \equiv (\alpha-1)\, \Gamma(\alpha-1). \tag{3.165}$$

If α is a positive integer, then it is possible through proof by induction to show that

$$\Gamma(\alpha) = (\alpha-1)!, \quad \alpha \in \mathbb{N}^+, \tag{3.166}$$

where $\mathbb{N}^+$ is the natural positive numbers. Some important properties of the gamma function are

$$\begin{aligned} \Gamma(\alpha) &= \pm\infty, & &\alpha = 0, -1, -2, \ldots, \\ \frac{1}{\Gamma(\alpha)} &= 0, & &\alpha = 0, -1, -2, \ldots, \\ \Gamma\left(\frac{1}{2}\right) &= \sqrt{\pi}, \\ \Gamma\left(\frac{5}{2}\right) &= \frac{3}{4}\sqrt{\pi}. \end{aligned}$$

Given the properties of the gamma function, it is possible to define the gamma distribution, which is a function of the gamma function, and the two parameters α and β. We therefore shall denote the gamma distribution by $Ga(\alpha, \beta)$, and it is defined as

$$Ga(\alpha, \beta) \equiv \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1} \exp\left\{-\frac{x}{\beta}\right\}, \tag{3.167}$$

where $\alpha, \beta > 0$, and the distribution is defined for $0 \leq x < \infty$. An important feature to note about the gamma distribution is the case where $\alpha = 1$. For this situation, the probability density function becomes

$$Ga(1, \beta) = \frac{1}{\beta} \exp\left\{-\frac{x}{\beta}\right\},$$

which is the definition of the exponential distribution.

Again, to help illustrate the shape and the structure of the gamma distribution, we have plotted four different combinations for the parameters α and β in Fig. 3.13. We can see the exponential distribution structure for the case where $\alpha = \beta = 1$, as just mentioned above, but for different combinations of the parameters we have quite different shapes and areas where the probabilities are being assigned.

3.11.1 Moment-Generating Function for the Gamma Distribution

The starting point for deriving the MGF for a gamma distribution is to consider the function

$$m_{Ga}(t) = \int_0^{\infty} \exp\{xt\} \frac{1}{\Gamma(\alpha)\beta^{\alpha}} x^{\alpha-1} \exp\left\{-\frac{x}{\beta}\right\} dx, \tag{3.168}$$

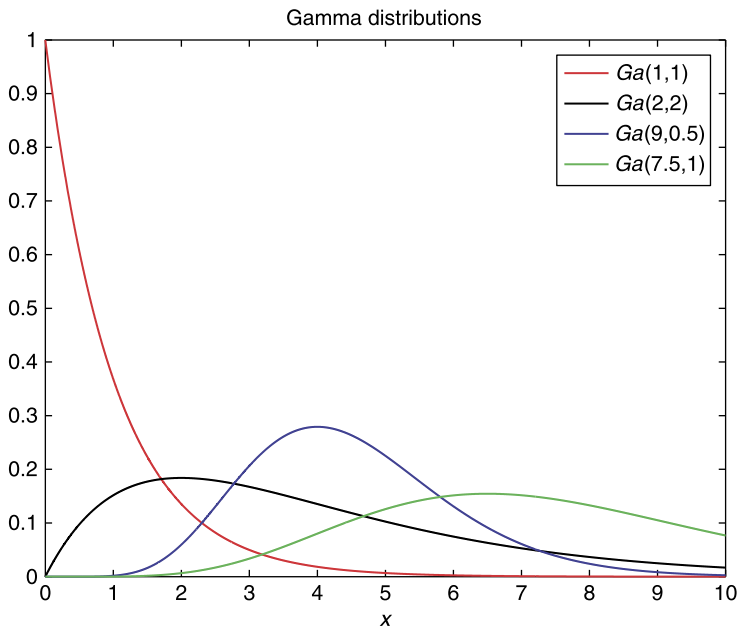


FIGURE 3.13

Plots of different versions of the gamma distribution.

where combining the exponentials and multiplying throughout by $\frac{(1-\beta t)^{\alpha}}{(1-\beta t)^{\alpha}}$ results in

$$m_{Ga}(t) = \frac{1}{(1-\beta t)^{\alpha}} \int_0^{\infty} \frac{x^{\alpha-1} \exp\left\{\frac{1-\beta t}{\beta}\right\}}{\Gamma(\alpha)\left(\frac{\beta}{1-\beta t}\right)^{\alpha}} dx.$$

While the expression in the integral above may look daunting, it is not. This is because the expression inside the integral is the definition of a gamma distribution with parameters $Ga\left(\alpha, \frac{\beta}{1-\beta t}\right)$, and therefore the integral of a gamma distribution's PDF is its CDF which integrates to one over all of its values; therefore the integral above is equal to one. As a result of this, we can define the MGF for a gamma distribution as

$$m_{Ga}(t) \equiv \frac{1}{(1-\beta t)^{\alpha}} = (1-\beta t)^{-\alpha}. \tag{3.169}$$

Differentiating (3.169) with respect to t and setting $t = 0$ yields the mean of the gamma distribution as

$$\begin{aligned} \mu &= \frac{d}{dt}(1-\beta t)^{-\alpha}\big|_{t=0} = \alpha\beta(1-\beta t)^{-(\alpha+1)}\Big|_{t=0} \\ &= \alpha\beta. \end{aligned} \tag{3.170}$$

As stated before, the variance can be found through finding the second non-central moment, which is equivalent to the second derivative of (3.169), evaluated at $t = 0$, and subtracting the square of (3.170). Therefore, the variance of the gamma distribution is

$$\begin{aligned} \sigma^2 &= \frac{d^2}{dt^2}(1-\beta t)^{-\alpha}\big|_{t=0} - \left(\frac{d}{dt}(1-\beta t)^{-\alpha}\big|_{t=0}\right)^2, \\ &= \alpha^2\beta^2 + \alpha\beta - \alpha^2\beta^2, \\ &= \alpha\beta^2. \end{aligned} \tag{3.171}$$

3.11.2 Skewness of the Gamma Distribution

We start the derivation of the skewness coefficient for the gamma distribution by recalling that we require the third non-central moment, which we know is equivalent to the third derivative of (3.169), evaluated at $t = 0$. Therefore, we have

$$\begin{aligned} \gamma_1 &= \frac{\mathbb{E}\left[X^3\right] - 3\mathbb{E}[X]\mathbb{E}\left[X^2\right] + (\mathbb{E}[X])^3}{\left(\mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2\right)^{\frac{3}{2}}}, \\ &= \frac{\alpha^3\beta^2 + 3\alpha^2\beta^3 + 2\alpha\beta^3 - 3\alpha^3\beta^3 - 3\alpha^2\beta^3 + 2\alpha^3\beta^3}{\alpha\beta^2\sqrt{\alpha\beta}}, \\ &= \frac{2}{\sqrt{\alpha}}. \end{aligned} \tag{3.172}$$

The importance of deriving the skewness coefficient for the gamma distribution is to illustrate that as $\alpha \to \infty$, the skewness tends to zero and as such the shape of the distribution becomes more symmetric,

similar to the Gaussian distribution but only for positive real numbers. It is therefore not exactly a Gaussian distribution, as it does not allow for negative values for the random variable. However, when α is sufficiently small, there is a skewness to the gamma distribution to the left and it is therefore a positively skewed distribution.

We now consider kurtosis for the gamma distribution to see how it is a function of the parameters of the gamma distribution.

3.11.3 Kurtosis of the Gamma Distribution

The starting point in the derivation of the kurtosis coefficient for the gamma distribution is to evaluate the fourth derivative of (3.169) at $t = 0$. This can easily be shown to be

$$\mathbb{E}\left[X^4\right] \equiv \frac{d^4}{dt^4}(1-\beta t)^{-\alpha}\bigg|_{t=0} = \left(\alpha^4 + 6\alpha^3 + 11\alpha^2 + 6\alpha\right)\beta^4.$$

Therefore, evaluating the expression for the coefficient of kurtosis

$$\gamma_2 = \frac{\mathbb{E}\left[X^4\right] - 4\mu\mathbb{E}\left[X^3\right] + 6\mu^2\mathbb{E}\left[X^2\right] - 3\mu^4}{\sigma^4},$$

yields the following expression for the gamma distribution:

$$\gamma_2 = \frac{\beta^4\left(\left[\alpha^4 + 6\alpha^3 + 11\alpha^2 + 6\alpha\right] - 4\left[\alpha^4 + 3\alpha^3 + 2\alpha^2\right] + 6\left[\alpha^4 + \alpha^3\right] - 3\alpha^4\right)}{\alpha^2\beta^4}.$$

After collecting the terms of like powers of α, it can easily be shown that the coefficient of kurtosis for the gamma distribution is

$$\gamma_2 = 3 + \frac{6}{\alpha}. \tag{3.173}$$

As with the coefficient of skewness, if we allow α to tend to infinity, we see that the coefficient of kurtosis defined in (3.173) tends to 3, which is the value for the Gaussian distribution. Therefore, if the data that you are modeling with a gamma distribution has a large value for α, then it is may be possible to approximate the data with a Gaussian distribution on the positive real number line.

3.11.4 Median of the Gamma Distribution

The definition for the median, as we know, is the value of the continuous random variable, x_{med}, such that

$$\int_a^{x_{med}} p(x)\,dx = 0.5.$$

Therefore, for the gamma distribution, we have

$$x_{med} \quad \text{s.t.} \quad \frac{1}{\Gamma(\alpha)\beta^\alpha}\int_0^{x_{med}} x^{\alpha-1}\exp\left\{-\frac{x}{\beta}\right\}dx = \frac{1}{2},$$

where, unlike the mean, cannot be solved analytically. It is however possible to find numerical approximations by using quadrature rules to numerically integrate the integral above.

3.11.5 Mode of the Gamma Distribution

We now consider the maximum likelihood state for the gamma distribution. As we have seen from the previous distribution, this state is found by differentiating the PDF and setting the derivative equal to zero. Therefore, the first derivative of (3.167) is

$$\begin{aligned}\frac{d}{dx}Ga(\alpha,\beta) &\equiv \frac{1}{\Gamma(\alpha)\beta^{\alpha}}\frac{d}{dx}\left(x^{\alpha-1}\exp\left\{-\frac{x}{\beta}\right\}\right)=0,\\ &= \frac{1}{\Gamma(\alpha)\beta^{\alpha}}\left((\alpha-1)x^{\alpha-2}\exp\left\{-\frac{x}{\beta}\right\}-\frac{1}{\beta}x^{\alpha-1}\exp\left\{-\frac{x}{\beta}\right\}\right)=0,\\ &\Rightarrow\left((\alpha-1)x^{\alpha-2}-\frac{1}{\beta}x^{\alpha-1}\right)=0,\\ &\Rightarrow x_{mode}=\beta(\alpha-1).\end{aligned} \tag{3.174}$$

Therefore, we have the property associated with positively skewed distributions that the mode is less than or equal to the median, and that the median is less than or equal to the mean.

3.11.6 Remarks About the Gamma Distribution and the Gaussian Distribution

An important property of the Gaussian distribution that we need to keep in mind is that all three descriptive statistics (mean, median, and mode) are the same, so if we consider the two descriptive statistics that we have analytical expressions for the gamma distribution, the mean and the mode, then we can see that there is a problem if we let $\alpha \to \infty$, as both the mean (and the variance) and the mode would tend to infinity. However, what is important about the expression for both the skewness and the kurtosis of the gamma distribution is that they are independent of β, while the mean, mode and variance are not; however, as we are about to demonstrate, if α is allowed to become very large, then in the limit as $\alpha \to \infty$, we see that the mean, mode, and variance are not similar to the Gaussian distribution's equivalent descriptive statistics. Therefore, as a caveat, the Gaussian approximation may hold for a reasonable large value for α, but is dependent on an associated value of β.

To verify the caveat about the values of α, we recall that the mean and the mode are $\alpha\beta$ and $\beta(\alpha-1)$, respectively. If k is becoming large, then so are the mean and the mode, but depending on the value of β they may not be converging to the same values. While the higher-order moments tend toward those of the Gaussian distribution, we still require the mean and the mode to also be converging to the same expression. If β is equal to 1 then α and $\alpha-1$ are approximately the same. If $\beta=\frac{1}{\alpha^2}$ then both the mean and the mode would be converging toward zero, with the mode converging faster than the mean due to the $\frac{1}{\alpha^2}$ term. However, the variance of the gamma distribution would be tending to zero faster than both the mean and the mode! Therefore, be cautious when using a Gaussian distribution approximation to gamma-distributed random variables, as there appears to be a finite range of values for α and β that the Gaussian approximation will hold. Also remember that the gamma distribution is defined for semi-positive definite random variables; that is to say it is defined for random variables that are greater than or equal to zero, but cannot not go negative, while the Gaussian distribution is defined for all real numbers.

3.11.7 Properties of Gamma-Distributed Random Variables

We have seen for the Gaussian distribution that if we have a set of Gaussian random variables that are independently distributed, then the sum of these random variables is also a Gaussian random variable. We have seen a geometric equivalent for the product of independently distributed lognormal random variables. For gamma-distributed independently distributed random variables, we have a similar additive property but there is a restriction that did not apply to the sum of independently distributed normal random variables. This restriction is that while the gamma random variables can have different shape parameters α_i, they must have the same scale parameter β. Therefore, for the sum of independently distributed gamma random variables, we have

$$Z = \sum_{i=1}^{n} X_i \sim Ga\left(\alpha_i, \beta\right) \Rightarrow Z \sim Ga\left(\sum_{i=1}^{n} \alpha_i, \beta\right). \tag{3.175}$$

3.12 Inverse Gamma Distribution

The inverse gamma distribution is part of an ensemble data assimilation system referred to as the gamma-inverse-gamma-Gaussian filter, or GIGG filter, [37]. Thus we present the univariate version here to prepare the understanding of the filter in a later chapter. We start by linking this distribution to the gamma distribution.

If a random variable X has a gamma distribution, $Ga\left(\alpha, \beta\right)$, then the random variable, $Y = \dfrac{1}{X}$, has an inverse gamma distribution, $IGa\left(\alpha, \dfrac{1}{\beta}\right)$. The definition for the inverse gamma distribution is given by

$$IGa\left(\alpha, \frac{1}{\beta}\right) \equiv \frac{\beta^{\alpha}}{\Gamma\left(\alpha\right)} x^{-\alpha-1} \exp\left\{-\frac{\beta}{x}\right\}. \qquad x > 0 \tag{3.176}$$

where α is the shape parameter, such that $\alpha > 0$, and β is the scale parameter, where $\beta > 0$. We have a plotted four different versions of the inverse-gamma PDF, for the situations: $\alpha = 1, \beta = 1, \alpha = 2, \beta = 1$, $\alpha = 3, \beta = 1$ and $\alpha = 3, \beta = 0.5$, in Fig. 3.14, where we can see the affects of increasing α, but holding β fixed, but also holding α fixed and halving the value of β. For $\beta = 1$, as α increases the peak of the distribution moves towards 1 and the distribution appears to be similar in appearance to a lognormal distribution. However, the difference in the plots for $\beta = 1$ and $\beta = 0.5$, for $\alpha = 3$, shows a narrowing of the peak, and an increase of the value of the PDF at the peak.

3.12.1 Moments of the Inverse-Gamma Distribution

We would normally introduce the moment generating function here, but one does not exist for the inverse gamma distribution. However, the integral expression for the expectation operator in this case is not too bad for the n-th non-central moment, and can be found through

$$\mathbb{E}\left[X^n\right] = \frac{\beta^{\alpha}}{\Gamma\left(\alpha\right)} \int_0^{\infty} x^n x^{-\alpha-1} \exp\left\{-\frac{\beta}{x}\right\} dx,$$

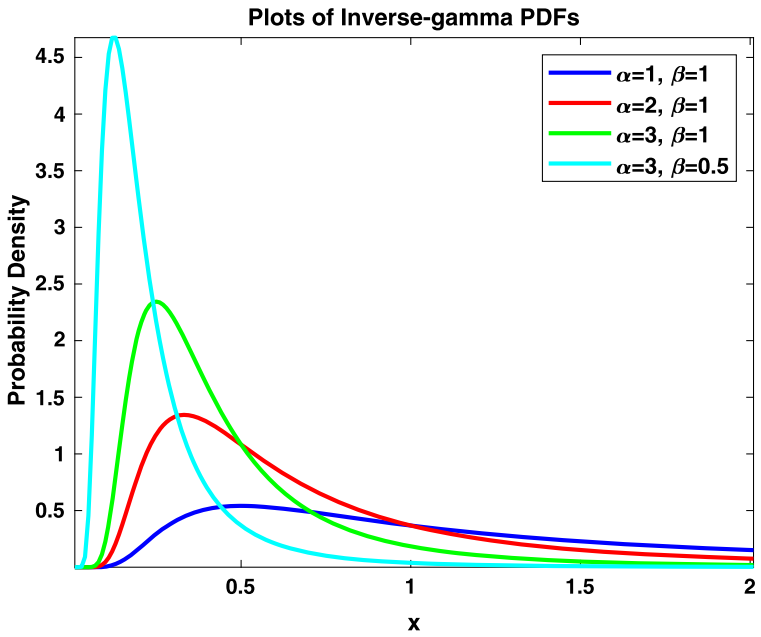


FIGURE 3.14

Plot of four different configurations of the inverse-gamma probability density function.

$$
\begin{aligned}
&= \frac{\beta^{\alpha}}{\Gamma(\alpha)} \int_0^{\infty} x^{n-\alpha-1} \exp\left\{-\frac{\beta}{x}\right\} dx, \\
&= \frac{\beta^{\alpha}}{\Gamma(\alpha)} \frac{\Gamma(\alpha-n)}{\beta^{\alpha-n}}, \\
&= \frac{\beta^{n}\Gamma(\alpha-n)}{(\alpha-1)\cdots(\alpha-n)\,\Gamma(\alpha-n)}, \\
&= \frac{\beta^{n}}{(\alpha-1)\cdots(\alpha-n)}.
\end{aligned} \tag{3.177}
$$

Thus the mean of the inverse-gamma distribution is

$$\mathbb{E}\left[X^1\right] = \frac{\beta}{\alpha-1}. \tag{3.178}$$

To find the variance we require the second order non-central moment, which can easily be shown to be

$$\mathbb{E}\left[X^2\right] = \frac{\beta}{(\alpha-1)(\alpha-2)}, \tag{3.179}$$

and so the variance is given by

$$VAR[X] \equiv \mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2 = \frac{\beta^2}{(\alpha-1)(\alpha-2)} - \frac{\beta^2}{(\alpha-1)^2} = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}. \tag{3.180}$$

An important feature to note about the variance in (3.180) is that it is only valid for values of $\alpha > 2$, otherwise we would be dividing by zero.

3.12.2 Skewness of the Inverse-Gamma Distribution

Recalling the definition of skewness we have

$$\beta_1 = \frac{\mathbb{E}\left[X^3\right] - 3\mathbb{E}[X]\mathbb{E}\left[X^2\right] + 2\mathbb{E}[X]^3}{\left(\mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2\right)\sqrt{\left(\mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2\right)}}.$$

Taking the numerator first we have

$$\begin{aligned}
\beta_{1,num} &= \frac{\beta^3}{(\alpha-1)(\alpha-2)(\alpha-3)} - \frac{3\beta^3}{(\alpha-1)^2(\alpha-2)} + \frac{2\beta^3}{(\alpha-1)^3}, \\
&= \beta^3\left(\frac{(\alpha-1)^5(\alpha-2) - 3(\alpha-1)^4(\alpha-2)^2(\alpha-3) + 2(\alpha-1)^2(\alpha-2)}{(\alpha-1)^6(\alpha-2)^2(\alpha-2)(\alpha-3)}\right), \\
&= \beta^3\left(\frac{(\alpha-1)^2 + 3(\alpha-1)(\alpha-3) + (\alpha-2)(\alpha-3)}{(\alpha-1)^3(\alpha-2)(\alpha-3)}\right), \\
&= \beta^3\left(\frac{\alpha^2 - 2\alpha + 1 - 6\alpha^2 + 12\alpha - 9 + 2\alpha^2 - 10\alpha + 12}{(\alpha-1)^3(\alpha-2)(\alpha-3)}\right), \\
&= \frac{4\beta^3}{(\alpha-1)^3(\alpha-2)(\alpha-3)}.
\end{aligned}$$

Turning out attention to the denominator we have

$$\beta_{1,den} = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}\sqrt{\frac{\beta^2}{(\alpha-1)^2(\alpha-2)}} = \frac{\beta^3}{(\alpha-1)^3(\alpha-2)\sqrt{(\alpha-2)}}.$$

Combining the numerator with the denominator results in

$$\begin{aligned}
\beta_1 = \frac{\beta_{1,num}}{\beta_{1,den}} &= \frac{\dfrac{4\beta^3}{(\alpha-1)^3(\alpha-2)(\alpha-3)}}{\dfrac{\beta^3}{(\alpha-1)^3(\alpha-2)\sqrt{(\alpha-2)}}}, \\
&= \frac{4\sqrt{(\alpha-2)}}{(\alpha-3)}.
\end{aligned} \tag{3.181}$$

We should note there that the skewness is independent of the scale parameter, and that this expression is only valid for $\alpha > 4$.

3.12.3 Kurtosis of the Inverse-Gamma Distribution

Recalling the definition for excess kurtosis we have;

$$\gamma_2 = \frac{\mathbb{E}\left[X^4\right] - 4\mathbb{E}[X]\mathbb{E}\left[X^3\right] + 6\mathbb{E}[X]^{22}\mathbb{E}\left[X^2\right] - 3\mathbb{E}[X]^4}{\left(\mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2\right)^2} - 3.$$

Forming the denominator first this time we have

$$\gamma_{1,den} = \frac{\beta^4}{(\alpha-1)^3(\alpha-2)^2}. \tag{3.182}$$

Now forming the numerator we have:

$$\gamma_{1,num} = \frac{\beta^4}{(\alpha-1)(\alpha-2)(\alpha-3)(\alpha-4)} - \frac{4\beta^4}{(\alpha-1)^2(\alpha-2)(\alpha-3)} + \frac{6\beta^4}{(\alpha-1)^3(\alpha-2)} - \frac{3\beta^4}{(\alpha-1)^4}. \tag{3.183}$$

While this expression may look difficult we can make one simplification through canceling an $(\alpha-1)$ from the denominator and the numerator. We can also remove all of the β^4 terms as well.

We are going to start combining the fractions, but first we will denote each of the 4 fractions in (3.183) as T_1 to T_4 from left to right. This is to enable us to keep track of the terms that we are combining. We start with combining T_3 with T_4 to form T_{34}:

$$\begin{aligned} T_{34} &= \frac{6(\alpha-1)^3 - 3(\alpha-1)^2(\alpha-2)}{(\alpha-1)^5(\alpha-2)}, \\ &= \frac{6(\alpha-1) - 3(\alpha-2)}{(\alpha-1)^3(\alpha-2)}. \end{aligned}$$

Combining T_2 with T_{34} to form T_{234} we have

$$\begin{aligned} T_{234} &= \frac{-4(\alpha-1)^3(\alpha-2) + 6(\alpha-1)^2(\alpha-2)(\alpha-3) - 3(\alpha-1)(\alpha-2)^2(\alpha-3)}{(\alpha-1)^4(\alpha-2)(\alpha-3)}, \\ &= \frac{-4(\alpha-1)^2 + 6(\alpha-1)(\alpha-3) - 3(\alpha-2)(\alpha-3)}{(\alpha-1)^3(\alpha-2)(\alpha-3)}. \end{aligned}$$

Forming the final sum, T_{1234}, we have

$$\begin{aligned} T_{1234} &= \frac{(\alpha-1)^3(\alpha-2)(\alpha-3) - 4(\alpha-1)^2(\alpha-2)(\alpha-3)(\alpha-4)}{(\alpha-1)^3(\alpha-2)^2(\alpha-3)^2(\alpha-4)} \\ &+ \frac{+6(\alpha-1)(\alpha-2)(\alpha-3)^2(\alpha-4) - 3(\alpha-2)^2(\alpha-3)^2(\alpha-4)}{(\alpha-1)^3(\alpha-2)^2(\alpha-3)^2(\alpha-4)} \\ &= \frac{(\alpha-1)^3 - 4(\alpha-1)^2(\alpha-4) + 6(\alpha-1)(\alpha-3)(\alpha-4) - 3(\alpha-2)(\alpha-3)(\alpha-4)}{(\alpha-4)^3(\alpha-2)(\alpha-3)(\alpha-4)} \end{aligned}$$

Multiplying out all of the brackets, dividing by the denominator, $\gamma_{1,den}$, and canceling terms results in the expression for the kurtosis of

$$\gamma_1 = \frac{(3\alpha+15)(\alpha-2)}{(\alpha-3)(\alpha-4)}. \tag{3.184}$$

To form the excess kurtosis we need to subtract 3 from (3.184), which yields

$$\begin{aligned}\gamma_2 &= \frac{(3\alpha+15)(\alpha-2)}{(\alpha-3)(\alpha-4)} - 3,\\ &= \frac{(3\alpha+15)(\alpha-2) - 3(\alpha-3)(\alpha-4)}{(\alpha-3)(\alpha-4)},\\ &= \frac{6(5\alpha-11)}{(\alpha-3)(\alpha-4)}.\end{aligned} \tag{3.185}$$

We now move on to the mode of the inverse-gamma distribution, noting that it appears that there is no expression for the median of the inverse-gamma distribution.

3.12.4 Mode of the Inverse-Gamma Distribution

As we ave seen many times now, the mode of a PDF is found through setting the first derivative to zero and then solving for the mode. Therefore, we have

$$\begin{aligned}x_{mode} &= \frac{d}{dx}\left(\frac{\beta^\alpha}{\Gamma(\alpha)}x^{-\alpha-1}\exp\left\{-\frac{\beta}{x}\right\}\right) = 0,\\ &= \frac{(-\alpha-1)\beta^\alpha}{\Gamma(\alpha)}x^{-\alpha-2}\exp\left\{-\frac{\beta}{x}\right\} + \frac{\beta^\alpha\beta}{\Gamma(\alpha)}x^{-\alpha-3}\exp\left\{-\frac{\beta}{x}\right\} = 0,\\ &= \beta - (\alpha+1)x = 0,\\ &= \frac{\beta}{\alpha+1}.\end{aligned} \tag{3.186}$$

As mentioned at the start of this section, the inverse-gamma distribution makes up part of the GIGG filter that we shall introduce in the non-Gaussian based data assimilation chapter. We now move on to a distribution that can also be linked to the gamma distribution; the beta distribution.

3.13 Beta Distribution

The beta distribution is similar to the gamma distribution in that it is defined in terms of a function, referred to as the **beta function**. The beta function is defined in terms of an integral, and is given by

$$B(\alpha,\beta) \equiv \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}\,dx. \tag{3.187}$$

There is a relationship that links the beta function to the gamma function, which is given by

$$B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}. \tag{3.188}$$

We shall now prove the relationship above between the gamma function and beta function. The starting point is to consider the definition of the product of two gamma functions, $\Gamma(\alpha)\Gamma(\beta)$, which is

$$\Gamma(\alpha)\Gamma(\beta) = \int_0^\infty\int_0^\infty x_1^{\alpha-1}x_2^{\beta-1}\exp\{-x_1\}\exp\{-x_2\}\,dx_1dx_2. \tag{3.189}$$

Next we introduce the change of variables

$$\begin{aligned} Y_1 &= u_1(X_1, X_2) = \frac{X_1}{X_1 + X_2}, & 0 \leq y_1 \leq 1, \\ Y_2 &= u_2(X_1, X_2) = X_1 + X_2, & 0 \leq y_2 < \infty. \end{aligned}$$

Inverting the expressions above to obtain X_1 and X_2 in terms of Y_1 and Y_2 results in

$$\begin{aligned} X_1 &= v_1(Y_1, Y_2) = Y_1 Y_2, \\ X_2 &= v_2(Y_1, Y_2) = (1 - Y_1) Y_2. \end{aligned}$$

Changing the integrating variables for a two-dimensional case requires the following formula:

$$g(y_1, y_2) = f(x_1, x_2) |J|,$$

where $|J|$ is the determinant of the Jacobian matrix which for a two variable change of variable transform is given by

$$|J| = \begin{vmatrix} \frac{\partial x_1}{\partial y_1} & \frac{\partial x_1}{\partial y_2} \\ \frac{\partial x_2}{\partial y_1} & \frac{\partial x_2}{\partial y_2} \end{vmatrix}. \tag{3.190}$$

Therefore, for our change of variable given above, we have

$$|J| = \begin{vmatrix} \frac{\partial x_1}{\partial y_1} & \frac{\partial x_1}{\partial y_2} \\ \frac{\partial x_2}{\partial y_1} & \frac{\partial x_2}{\partial y_2} \end{vmatrix} = \begin{vmatrix} y_2 & y_1 \\ -y_2 & 1 - y_1 \end{vmatrix} = y_2(1 - y_1) + y_1 y_2 = y_2.$$

Substituting the information above into (3.189),

$$\begin{aligned} g(y_1, y_2) &= \int_0^1 \int_0^\infty (y_1 y_2)^{\alpha - 1} [(1 - y_1) y_2]^{\beta - 1} \exp\{-y_1 y_2 + (1 - y_1) y_2\} y_2 dy_1 dy_2, \\ &= \int_0^1 \int_0^\infty y_1^{\alpha - 1} (1 - y_1)^{\beta - 1} y_2^{\alpha + \beta - 1} \exp\{-y_2\} dy_1 dy_2, \\ &= \int_0^1 y_1^{\alpha - 1} (1 - y_1)^{\beta - 1} dy_1 \int_0^\infty y_2^{\alpha + \beta - 1} \exp\{-y_2\} dy_2, \\ &= B(\alpha, \beta) \Gamma(\alpha + \beta), \end{aligned}$$

but the original function was $g(y_1, y_2) = \Gamma(\alpha) \Gamma(\beta)$; we therefore have

$$B(\alpha, \beta) = \frac{\Gamma(\alpha) \Gamma(\beta)}{\Gamma(\alpha + \beta)}.$$

Given the definition of the beta function, it is possible to define beta distribution for the beta-distributed random variable where $X \in (0, 1)$, as

$$Beta(X) = \frac{1}{B(\alpha, \beta)} x^{\alpha - 1} (1 - x)^{\beta - 1}, \tag{3.191}$$

which can be expressed in terms of gamma functions as

$$Beta\,(\alpha,\beta)=\frac{\Gamma\,(\alpha+\beta)}{\Gamma\,(\alpha)\,\Gamma\,(\beta)}x^{\alpha-1}\,(1-x)^{\beta-1}\,,$$

where $\alpha,\beta>0$.

The beta distribution can be linked to the gamma distribution through considering the two gamma-distributed random variables, $X_1\sim Ga\,(\alpha,\theta)$, and $X_2\sim Ga\,(\beta,\theta)$, where the new random variable $\frac{X_1}{X_1+X_2}$ is a beta-distributed random variable, $Beta\,(\alpha,\beta)$. The beta distribution can also be linked to the continuous uniform distribution with $Beta\,(1,1)$.

An important property of the beta distribution is that $X_1\sim Beta\,(\alpha,\beta)$ is the mirror image of the beta distribution for $X_2\sim Beta\,(\beta,\alpha)$.

Before we move onto the standardized moments of the beta distribution, we consider the effects that the parameters have on the shape of the distribution. In Fig. 3.15 we have plotted four different beta distributions, where it is clear that there are some very different shapes to the distribution compared to those that we have seen with the previous four. When $\alpha=\beta=2$, the blue curve in Fig. 3.15, then the distribution appears to be slightly similar to a lognormal distribution but only valid for $x\in(0,1)$. The curve associated with $\alpha=2$ and $\beta=5$ appears to be semi-circular. Finally, an important feature to notice about the beta distribution is that it changes skewness from left-skewed to right-skewed, which we have not seen with the previous distributions.

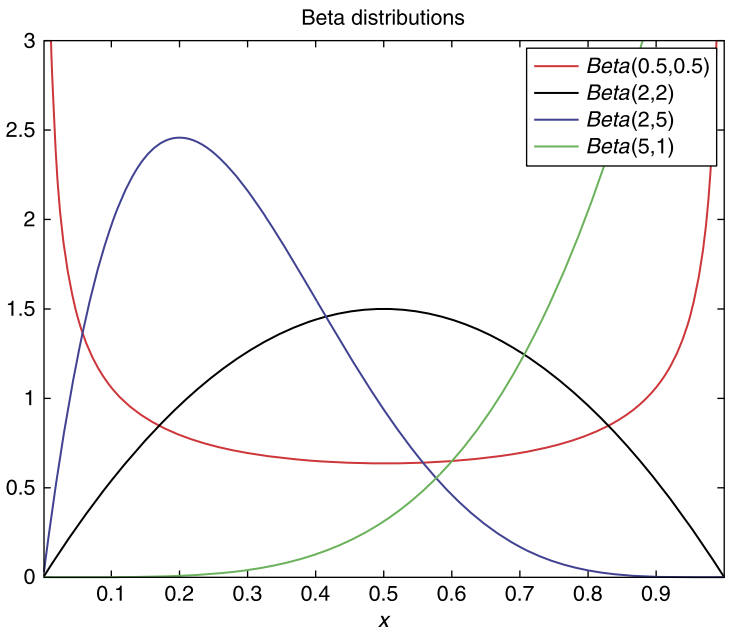


FIGURE 3.15

Plots of different versions of the beta distribution.

3.13.1 Moments of the Beta Distribution

The first feature to note about the beta distribution, like the lognormal distribution, is that a moment-generating function does not exist for this distribution. Therefore, to find the mean we must evaluate the expectation operator for the beta distribution, which is

$$\mu = \mathbb{E}[X] = \int_0^1 \frac{1}{B(\alpha, \beta)} x x^{\alpha-1} (1-x)^{\beta-1} \, dx. \tag{3.192}$$

The terms involving the random variable in (3.192) is the definition of a beta function $B(\alpha+1, \beta)$, so at this point we can express the mean of the beta distribution in terms of beta functions as

$$\mu = \frac{B(\alpha+1, \beta)}{B(\alpha, \beta)}, \tag{3.193}$$

which is still an expression involving integrals. To simplify further, we use the following properties of the beta and gamma functions

$$B(\alpha, \beta) \equiv \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}, \qquad \Gamma(\alpha) \equiv (\alpha-1)\Gamma(\alpha-1).$$

This enables (3.193) to be rewritten as

$$\begin{aligned} \mu &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \cdot \frac{\Gamma(\alpha+1)\Gamma(\beta)}{\Gamma(\alpha+\beta+1)}, \\ &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \cdot \frac{\alpha\Gamma(\alpha)\Gamma(\beta)}{(\alpha+\beta)\Gamma(\alpha+\beta)}, \\ &= \frac{\alpha}{\alpha+\beta}. \end{aligned} \tag{3.194}$$

To find the variance of the beta distribution, we require the second order non-central moment, which can be shown to be

$$\mu_2' \equiv \mathbb{E}\left[X^2\right] = \frac{\alpha(\alpha+1)}{(\alpha+\beta)(\alpha+\beta+1)}. \tag{3.195}$$

This makes the definition for the variance of the beta distribution as

$$\sigma^2 = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}. \tag{3.196}$$

The third order non-central moment can be shown to be

$$\mu_3' \equiv E\left[X^3\right] = \frac{(\alpha+2)(\alpha+1)\alpha}{(\alpha+\beta)(\alpha+\beta+1)(\alpha+\beta+2)}, \tag{3.197}$$

which leads to expression for the skewness of the beta distribution as

$$\beta_1 \equiv \frac{2(\beta-\alpha)\sqrt{\alpha+\beta-1}}{(\alpha+\beta+2)\sqrt{\alpha\beta}}. \tag{3.198}$$

Exercise 3.46. *Verify the expression for the third order non-central moment of the beta distribution in* (3.197) *and use this to derive the coefficient of skewness given in* (3.198).

As with the other distributions we have presented, we now consider the fourth order non-central moment, which is required to find the coefficient of kurtosis. Given the expressions that have been derived for the first, second, and third order non-central moments, then there appears to be a pattern involving the order of the moment and the expressions for the non-central moments. Therefore, the nth non-central moment for the beta distribution can be expressed as

$$\mu'_n \equiv \frac{\prod_{i=0}^{n-1}(\alpha+i)}{\prod_{i=0}^{n-1}(\alpha+\beta+i)}.$$

This implies that the fourth order non-central moment is given by

$$\mu'_4 = \mathbb{E}\left[X^4\right] = \frac{(\alpha+3)(\alpha+2)(\alpha+1)\alpha}{(\alpha+\beta+3)(\alpha+\beta+2)(\alpha+\beta+1)(\alpha+\beta)}.$$

Given the definition above, then the excess kurtosis for the beta distribution can be shown to be

$$\gamma_2 = \frac{6\left[(\alpha+\beta)^2(\alpha+\beta+1) - \alpha\beta(\alpha+\beta+2)\right]}{\alpha\beta(\alpha+\beta+2)(\alpha+\beta+3)}. \tag{3.199}$$

Exercise 3.47. *Verify that the expression in* (3.199) *is the excess kurtosis coefficient for the generalized beta distribution.*

3.13.2 Median of the Beta Distribution

The median of the beta distribution is found through determining the state, x_{med}, such that

$$x_{med} \quad \text{s.t.} \quad \frac{1}{\beta(\alpha,\beta)}\int_0^{x_{med}} x^{\alpha-1}(1-x)^{\beta-1}\,dx = \frac{1}{2}. \tag{3.200}$$

However, as with the gamma distribution, it is not possible to find an analytical solution to (3.200), and so this integral must be solved through numerical techniques.

3.13.3 Mode of the Beta Distribution

As with the other distributions, the modal value is obtained by setting the first derivative of the probability density function to zero. Therefore, the first derivative of (3.191) equal to zero is

$$\begin{aligned}
x_{mode} \equiv \left.\frac{dBeta(\alpha,\beta)}{dx}\right|_{x=x_{mode}} &= 0, \\
\Rightarrow \frac{1}{B(\alpha,\beta)}\frac{d}{dx}\left(x^{\alpha-1}(1-x)^{\beta-1}\right) &= 0, \\
\Rightarrow (\alpha-1)x^{\alpha-2}(1-x)^{\beta-1} - x^{\alpha-1}(\beta-1)(1-x)^{\beta-2} &= 0, \\
\Rightarrow x^{\alpha-1}(1-x)^{\beta-1}\left[(\alpha-1)x^{-1} - (\beta-1)(1-x)^{-1}\right] &= 0 \\
\Rightarrow \frac{(\alpha-1)}{x} - \frac{(\beta-1)}{1-x} &= 0, \\
\Rightarrow x_{mode} = \frac{\alpha-1}{\alpha+\beta-2}.
\end{aligned} \tag{3.201}$$

3.14 Chi-Squared (χ^2) Distribution

It is quite common to denote a continuous random variable that has a chi-squared distribution with n degrees of freedom by χ^2_n. The associated PDF for the χ^2_n distribution is given by

$$\chi^2_n = \frac{1}{2^{\frac{n}{2}}\Gamma\left(\frac{n}{2}\right)} \exp\left\{-\frac{x}{2}\right\} x^{\frac{n}{2}-1}, \quad x \geq 0. \tag{3.202}$$

We have plotted the distributions for $n = 1, 2, 3, 4, 5$ in Fig. 3.16. There are stark differences between the five distributions, but all appear to be left-skewed. However, unlike the lognormal distribution, it appears that the modes are increasing as n increases.

3.14.1 Moments of the Chi-Squared Distribution

Unlike with the gamma and beta distributions, a moment-generating function does exists for the chi-squared distribution. The starting point in the derivation of the MGF is the integral

$$M_{\chi^2}(t) \equiv \mathbb{E}\left[\exp\{tx\}\right] = \frac{1}{2^{\frac{n}{2}}\Gamma\left(\frac{n}{2}\right)} \int_0^\infty \exp\{tx\} \exp\left\{-\frac{x}{2}\right\} x^{\frac{n}{2}-1} dx. \tag{3.203}$$

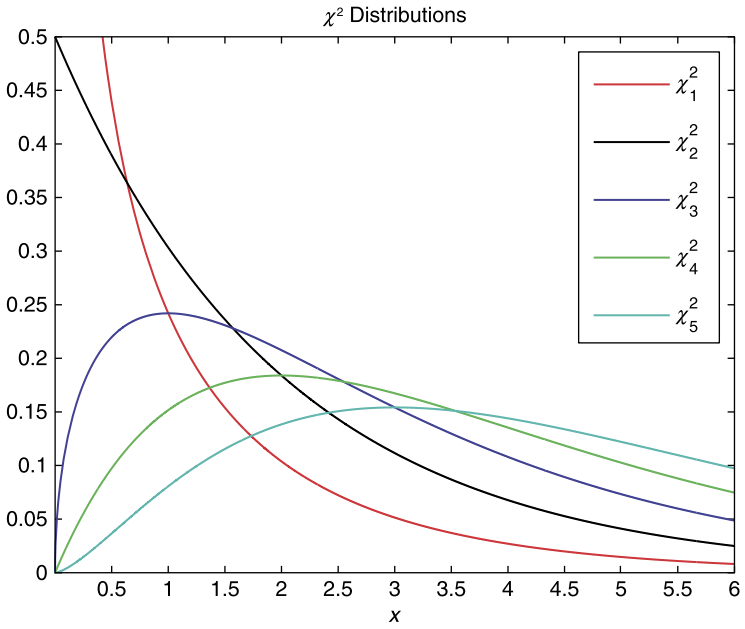


FIGURE 3.16

Plots of different versions of the χ^2 distribution.

Combining the exponential above enables us to simplify (3.203) to

$$M_{\chi^2}(t) = \frac{1}{2^{\frac{n}{2}}\Gamma\left(\frac{n}{2}\right)} \exp\left\{-\frac{x}{2}(1-2t)\right\} x^{\frac{n}{2}-1} dx. \tag{3.204}$$

An important feature to note here is that the integral above is only valid for values of $t < \frac{1}{2}$, otherwise the exponential in (3.204) is a growing exponential and therefore not bounded. To solve the integral in (3.204), we introduce the change of variable

$$x = \frac{u}{\frac{1}{2}-t} \Rightarrow u = \left(\frac{1}{2}-t\right)x \Rightarrow \mathrm{dx} = \frac{du}{\left(\frac{1}{2}-t\right)}, \qquad \begin{array}{l} x = 0 \Rightarrow u = 0, \\ x = \infty \Rightarrow u = \infty. \end{array} \tag{3.205}$$

Therefore, substituting all the information from (3.205) into (3.204), we obtain

$$M_{\chi^2}(t) = \frac{1}{2^{\frac{n}{2}}} \frac{1}{\left(\frac{1}{2}-t\right)^{\frac{n}{2}}} \frac{1}{\Gamma\left(\frac{n}{2}\right)} \int_0^\infty \exp\{-u\} u^{\frac{n}{2}-1} dx. \tag{3.206}$$

The integral in (3.206) is the definition of the gamma function for $\frac{n}{2}$, i.e., $\Gamma\left(\frac{n}{2}\right)$, which cancels the term in the denominator. It is then possible to combine the remaining two terms as they are to the same power. Therefore, the moment-generating function for the chi-squared distribution is

$$M_{\chi^2}(t) = (1-2t)^{-\frac{n}{2}}, \quad t < \frac{1}{2}. \tag{3.207}$$

To find the first four non-central moments of the chi-squared distribution requires differentiating (3.207) to the fourth order, and evaluating these derivatives at $t = 0$, which results in

$$\begin{aligned} \mu_1 &= n, \\ \mu_2' &= n^2 + 2n, \\ \mu_3' &= n^3 + 6n^2 + 8n, \\ \mu_4' &= n^4 + 12n^3 + 44n^2 + 48n. \end{aligned} \tag{3.208}$$

Given the non-central moments in (3.208) for the chi-squared distribution, it is possible to derive the mean, variance, coefficient of skewness, and coefficient of kurtosis for this distribution as

$$\begin{aligned} \mu &= n, \\ \sigma^2 &= 2n, \\ \beta_1 &= \sqrt{\tfrac{8}{n}}, \\ \beta_2 &= 3 + \tfrac{12}{n}. \end{aligned} \tag{3.209}$$

Exercise 3.48. *Verify, using the moment-generating function defined in* (3.207)*, the non-central moments in* (3.208)*, and then verify that the expressions for the mean, variance, coefficient of skewness, and coefficient of kurtosis for the chi-squared distribution in* (3.209) *are correct.*

An important observation to make here about the expressions for the coefficients of skewness and kurtosis is that as $n \to \infty$, then these standardized moments tend toward those of the Gaussian distribution.

3.14.2 Median of the Chi-Squared Distribution

As with the gamma and beta distributions, there is no analytical expression for the median of the chi-squared distribution. We know that it is the state x_{med} such that

$$x_{med} \quad \text{s.t.} \quad \frac{1}{2^{\frac{n}{2}}\Gamma\left(\frac{n}{2}\right)} \int_0^{x_{med}} \exp\left\{-\frac{x}{2}\right\} x^{\frac{n-2}{2}} dx = \frac{1}{2}. \tag{3.210}$$

There are several different approximations for the median of the chi-squared distribution, but we shall not go into detail about them here. One fact we know for certain is that it will be between the expression for the mode and the mean.

3.14.3 Mode of the Chi-Squared Distribution

To obtain the mode of the chi-squared we need to differentiate (3.202) and find the value x_{mode} such that the derivative is equal to zero at that state. Therefore, we have

$$\frac{d\chi_n^2}{dx} = \exp\left\{-\frac{x}{2}\right\}\left(-\frac{1}{2}x^{\frac{n}{2}-1} + \left(\frac{n}{2}-1\right)x^{\frac{n}{2}-2}\right) = 0.$$

Canceling the $x^{\frac{n}{2}-2}$ terms and rearranging results in the mode of the chi-square distribution being

$$x_{mode} = n - 2. \tag{3.211}$$

However, as we just mentioned, the expression in (3.211) for the mode of the chi-square distribution is *an* expression for the mode. The chi-square distribution is defined for semi-positive definite values, which means that when n is greater than or equal to two, the mode is equal to zero as the chi-square distribution is not defined for negative values. This fact about the mode implies that the structure of the distribution is heavily right-skewed.

Properties of the chi-squared distribution

The chi-squared distribution is an additive distribution which we shall now prove. If we have two independent chi-squared distributed continuous random variables, $X_1 \sim \chi_{n_1}^2$ and $X_2 \sim \chi_{n_2}^2$, then the random variable $Z = X_1 + X_2$ is also a chi-squared distributed random variable with the distribution $\chi_{n_1+n_2}^2$.

Proof. We start by recalling that the property of moment-generating functions for the sum of two independently distributed random variables is

$$\mathbb{E}\left[\exp\{(X_1 + X_2)\, t\}\right] \equiv \mathbb{E}\left[\exp\{X_1\}\right]\mathbb{E}\left[\exp\{X_2\}\right].$$

Substituting the moment-generating functions for the chi-squared distributions for X_1 and X_2 results in

$$\mathbb{E}\left[\exp\{Yt\}\right] = (1-2t)^{-\frac{n_1}{2}}\,(1-2t)^{-\frac{n_2}{2}} = (1-2t)^{-\frac{n_1+n_2}{2}}. \tag{3.212}$$

The expression in (3.212) is that for the moment-generating function of a chi-squared distribution, $\chi^2\,(n_1 + n_2)$, which by the uniqueness theorem can only be that of the chi-squared distribution.

Given the proof above, the result can be extended to the sum of N independently distributed chi-square random variables, $\sum_{i=1}^{N} X_i \sim \chi^2_{\left(\sum_{i=1}^{N} n_i\right)}$, where the proof is through induction.

3.14.4 Relationships to Other Distributions

- The most recognized relation of the chi-squared distribution to another distribution is that to the Gaussian distribution. If we have a random variable X which is a unit Gaussian distributed random variable, $X \sim N(0, 1)$ for $-\infty < x < \infty$, then $Z = X^2$ is a chi-squared distributed random variable with the distribution

$$f(z) = \frac{1}{\sqrt{2\pi}} x^{\frac{1}{2}} \exp\left\{-\frac{z}{2}\right\}, \quad z \geq 0 \equiv \chi_1^2.$$

  This result can be extended to the sum of N independent Gaussian distributed random variables where $\sum_{i=1}^{N} X_i \sim \chi_n^2$. These results do not depend on the original random variable being unit Gaussian distributed, but does require the variable to be transformed to the standardized Gaussian distribution through the transform $Z = \frac{X-\mu}{\sigma}$.
- Given that the definition of the χ_n^2 distribution contains a gamma function, it would appear that there could be a relationship to the gamma distribution. In fact, in the definition of the gamma distribution, if $\alpha = \frac{n}{2}$ and $\beta = 2$, then the gamma distribution is equivalent to the χ_n^2 distribution.
- The chi-squared distribution can also be related to the exponential distribution. If $X \sim \chi_2^2$, then X is also an exponentially distributed random variable with the distribution $X \sim Exp\left(\frac{1}{2}\right)$.
- If we have two independently distributed chi-squared distributed random variables, $X_1 \sim \chi_{n_1}^2$ and $X_2 \sim \chi_{n_2}^2$, then the new random variable $Z \equiv \frac{X_1}{X_1+X_2}$ is a beta-distributed random variable such that $Z \sim Beta\left(\frac{n_1}{2}, \frac{n_2}{2}\right)$.
- If X is a continuous uniform distributed random variable such that $x \sim U(0, 1)$, then the transformed random variable $Y = -2 \ln X$ is a chi-squared distributed random variable, $Y \sim \chi_2^2$.
- An important property of the chi-squared distribution is its use for determining goodness of fit between expected outcomes and observations. If we have a vector of Gaussian distributed random variables, $\boldsymbol{x}$ of length n (we shall go into multivariate distributions in the next chapter), a vector of means $\boldsymbol{\mu}$, and a covariance matrix $\boldsymbol{\Sigma}$ that is a positive definite matrix, then the random variable

$$X = (\boldsymbol{x} - \boldsymbol{x})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{x}),$$

  is a chi-squared distributed random variable with a χ_n^2 distribution.

3.15 Rayleigh Distribution

The Rayleigh distribution was originally derived by Lord Rayleigh, who is also referred to as J. W. Strutt in connection with a problem in acoustics. A Rayleigh distribution can often be observed when the overall magnitude of a vector is related to its directional components. An example where the Rayleigh distribution arises is when wind velocity is analyzed into its orthogonal two-dimensional vector components. Assuming that each component is an uncorrelated, Gaussian distributed random variable with equal variance, and zero mean, then the overall wind speed can be characterized by a Rayleigh distribution. This distribution is defined for values of $x \geq 0$, so it is therefore a semi-positive definite

distribution. The definition of the Rayleigh distribution is

$$Ra\left(\sigma^2\right) \equiv \frac{x}{\sigma^2}\int_0^{\infty} \exp\left\{-\frac{x^2}{2\sigma^2}\right\}, \quad 0 \leq x\infty,\ \sigma > 0. \tag{3.213}$$

We have plotted five different versions of the Rayleigh distribution in Fig. 3.17 for $\sigma^2 = 0.5, 1, 2, 3, 4$. We can see that where $\sigma^2 = 0.5$, the Rayleigh distribution appears to be quite similar to a lognormal distribution but does not have the steepness of function to the left of the mode as the lognormal distribution does. The same appears to be true for the case where $\sigma^2 = 1$. However, as the values for σ^2 increase, the distribution starts to flatten, but is still left skewed.

3.15.1 Moment-Generating Function for the Rayleigh Distribution

The moment-generating function for the Rayleigh distribution is quite a complicated expression, but we shall derive it here. The starting point is the definition for the moment-generating function:

$$\mathbb{E}\left[e^{xt}\right] \equiv \int_0^{\infty} e^{xt} x e^{-\frac{x^2}{2\sigma^2}} dx. \tag{3.214}$$

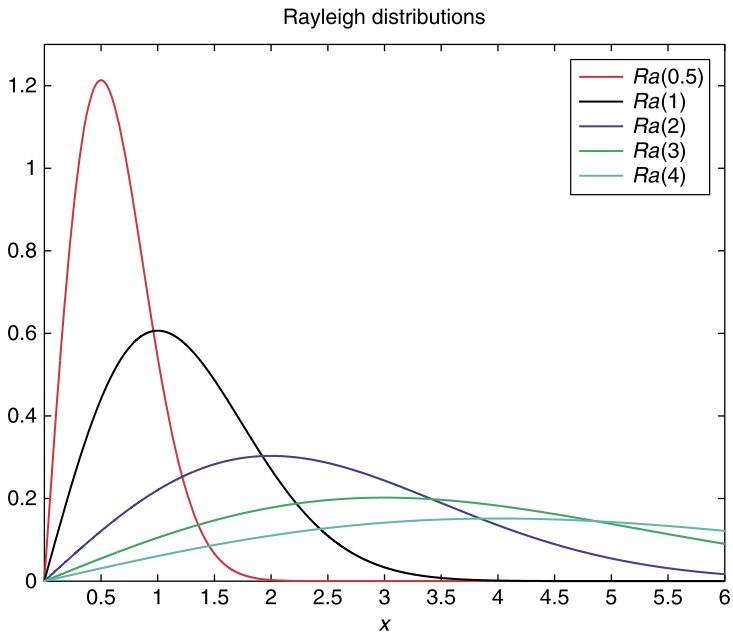


FIGURE 3.17

Plots of different versions of the Rayleigh distribution.

Completing the square for the exponentials results in

$$\mathbb{E}\left[e^{xt}\right] = e^{\frac{\sigma^2 t^2}{2}} \int_0^\infty x e^{-\left(\frac{x-\sigma^2 t}{\sqrt{2}\sigma}\right)^2} dx.$$

The next step is to introduce the change of variable

$$\omega = \frac{x-\sigma^2 t}{\sqrt{2}\sigma} \Rightarrow x = \sqrt{2}\sigma\omega + \sigma^2 t \Rightarrow \begin{array}{ll} x=0 & \omega = -\frac{\sigma^2 t}{\sqrt{2}} \\ x=\infty & \omega = \infty \end{array} \Rightarrow dx = \sqrt{2}\sigma d\omega. \tag{3.215}$$

Substituting all the information from (3.215) into (3.214) results in

$$\begin{aligned} \mathbb{E}\left[e^{xt}\right] &= e^{\frac{\sigma^2 t^2}{2}} \frac{1}{\sigma^2} \int_{-\frac{\sigma t}{\sqrt{2}}}^\infty \left(\sqrt{2}\sigma\omega + \sigma^2 t\right) e^{-\omega^2} \sqrt{2}\sigma d\omega, \\ &= e^{\frac{\sigma^2 t^2}{2}} \left(\underbrace{\int_{-\frac{\sigma t}{\sqrt{2}}}^\infty 2\omega e^{-\omega^2} d\omega}_{\textbf{(1)}} + \underbrace{\sqrt{2}\sigma t \int_{-\frac{\sigma t}{\sqrt{2}}}^\infty \sqrt{2}\sigma t e^{-\omega^2} d\omega}_{\textbf{(2)}} \right). \end{aligned} \tag{3.216}$$

Taking each term in order in (3.216), evaluating the integral in **(1)** first results in

$$e^{\frac{\sigma^2 t^2}{2}} \left[-e^{-\omega^2}\right]_{-\frac{\sigma t}{\sqrt{2}}}^\infty = e^{\frac{\sigma^2 t^2}{2}} \left(0 + e^{-\frac{\sigma^2 t^2}{2}}\right) = 1.$$

To evaluate the integral in **(2)**, we notice that the function in the integral is symmetric about $\omega = 0$, which means that it is possible to rewrite the integral as:

$$\sqrt{2}\sigma t \int_{-\frac{\sigma t}{\sqrt{2}}}^\infty \sqrt{2}\sigma t e^{-\omega^2} d\omega \equiv \sqrt{2}\sigma t \left(\int_0^{\frac{\sigma t}{\sqrt{2}}} e^{-\omega^2} d\omega + \int_0^\infty e^{-\omega^2} d\omega \right). \tag{3.217}$$

To evaluate the integrals in (3.217), we have to introduce the **error function, or erf**:

$$erf\,(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-\varepsilon^2} d\varepsilon. \tag{3.218}$$

Therefore, it is possible to rewrite (3.217) in terms of the error function as

$$\sqrt{2}\sigma t \left(\frac{\sqrt{\pi}}{2} erf\left(\frac{\sigma t}{\sqrt{2}}\right) + \int_0^\infty e^{-\omega^2} d\omega \right). \tag{3.219}$$

A property of the error function is that $erf\,(\infty) = 1$, which implies that the integral $\int_0^\infty e^{-\omega^2} d\omega = \frac{\sqrt{\pi}}{2}$. Thus the final expression for the MGF for the Rayleigh distribution is

$$M_{Ra}\,(t) = 1 + e^{-\frac{\sigma^2 t^2}{2}} \sigma t \sqrt{\frac{\pi}{2}} \left(erf\left(\frac{\sigma t}{\sqrt{2}}\right) + 1 \right). \tag{3.220}$$

3.15.2 Moments of the Rayleigh Distribution

In the previous subsection, we derived the MGF for the Rayleigh distribution, which is a function of the error function. To avoid complicated differentiation, we shall derive the non-central and standardized moments for the Rayleigh through the integration approach. Thus the mean of the Rayleigh distribution is found through evaluating the integral

$$\mu_1 = \mathbb{E}[X] = \int_0^\infty \frac{x^2}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx, \tag{3.221}$$

which can be solved by applying integration by parts, where

$$u = x \qquad v' = \frac{x}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\},$$
$$u' = 1 \qquad v = -\exp\left\{-\frac{x^2}{2\sigma^2}\right\}.$$

Combining the information above into the integration by parts formula yields

$$\mu = \left[-x \exp\left\{-\frac{x^2}{2\sigma^2}\right\}\right]_0^\infty + \int_0^\infty \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx.$$

As we have seen before, the first term in the expression above tends to zero, which leaves the integral, which is similar to that of the expectation of the standard Gaussian distribution but without the $\frac{1}{\sqrt{2\pi}\sigma}$ coefficient. Therefore, introducing the change of variable $t = \frac{x}{\sigma}$, implies that $dx = \sigma dx$, which then enables the integral above to be rewritten as

$$\mu = \sigma \int_0^\infty \exp\left\{-\frac{t^2}{2}\right\} dt \equiv \sigma \frac{\sqrt{2\pi}}{2} = \sigma\sqrt{\frac{\pi}{2}}. \tag{3.222}$$

As with the other distributions presented so far, we need to derive the non-central second order moment for the Rayleigh distribution to obtain the expression for the variance of this distribution. The second order non-central moment for this distribution is found through evaluating the integral

$$\mu_2' = \mathbb{E}\left[X^2\right] = \int_0^\infty \frac{x^3}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx. \tag{3.223}$$

Integrating by parts the integral in (3.223), we have

$$u = x^2, \qquad v' = \frac{x}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\},$$
$$u' = 2x, \qquad v = -\exp\left\{-\frac{x^2}{2\sigma^2}\right\},$$

which leads to

$$\mu_2' = \left[-x^2 \exp\left\{-\frac{x^2}{2\sigma^2}\right\}\right]_0^\infty + 2\int_0^\infty x \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx.$$

The first term above tends to zero, but we need to use integration by parts again for the integral, where this time we have

$$u = 1, \qquad v' = 2x \exp\left\{-\frac{x^2}{2\sigma^2}\right\},$$
$$u' = 0, \qquad v = -2\sigma^2 \exp\left\{-\frac{x^2}{2\sigma^2}\right\}.$$

This leads to the second order non-central moment for this distribution as

$$\mu_2' = \left[-2\sigma^2 \exp\left\{-\frac{x^2}{2\sigma^2}\right\}\right]_0^\infty + \int_0^\infty 0 \times 2\sigma^2 \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx.$$

Therefore, the integral is zero, and we are left with evaluating the limits on the first term, which results in

$$\mu_2' = 2\sigma^2. \tag{3.224}$$

Substituting (3.224) into the variance definition yields

$$Var[X] = 2\sigma^2 - \frac{\pi}{2}\sigma^2 \equiv \sigma^2\left(\frac{4-\pi}{2}\right). \tag{3.225}$$

3.15.3 Skewness of the Rayleigh Distribution

Moving on to the third order non-central moment, we need to evaluate

$$\mu_3' \equiv \mathbb{E}\left[X^3\right] = \int_0^\infty \frac{x^4}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx. \tag{3.226}$$

Because it takes multiple uses of integration by parts to evaluate (3.226), we shall leave the derivation as an exercise, but we can say that $\mu_3' = 3\sqrt{\frac{\pi}{2}}\sigma^3$.

Exercise 3.49. *Verify that* $\int_0^\infty \frac{x^4}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx = 3\sqrt{\frac{\pi}{2}}\sigma^3$.

It follows from the expressions for the third order non-central moment of the Rayleigh distribution that the skewness coefficient for this distribution is

$$\beta_1 = \frac{3\sqrt{\frac{\pi}{2}}\sigma^3 - 6\sqrt{\frac{\pi}{2}}\sigma^3 + \pi\sqrt{\frac{\pi}{2}}\sigma^3}{\sigma^3\left(\frac{4-\pi}{2}\right)\sqrt{\left(\frac{4-\pi}{2}\right)}},$$
$$= \frac{2\sqrt{\pi}\,(\pi-3)}{(4-\pi)^{\frac{3}{2}}}. \tag{3.227}$$

A notable feature in (3.227) for the coefficient of skewness for the Rayleigh distribution is that it is independent of both the shape and scale parameters. Therefore, the coefficient of skewness for a

Rayleigh distribution is a constant and it is the same for all choices of the parameters (α, β). The actual constant is approximately $\beta_1 \approx 0.631$.

3.15.4 Kurtosis of the Rayleigh Distribution

Finally moving onto the coefficient of kurtosis for the Rayleigh distribution, we require the fourth order non-central moment of the Rayleigh distribution, which can be shown, after a few applications of integrating by parts, as $\mu_4' \equiv E\left[X^4\right] = 8\sigma^4$. Therefore, the coefficient of kurtosis can be shown to be

$$\beta_2 = \frac{8\sigma^4 - \frac{3}{4}\pi^2\sigma^4}{\frac{(4-\pi)^2}{4}\sigma^4} = \frac{32 - 3\pi^2}{(4-\pi)^2}. \tag{3.228}$$

Exercise 3.50. *Verify that the fourth order non-central moment of the Rayleigh distribution,* $\mu_4' \equiv E\left[X^4\right]$*, is* $\mu_4' = 8\sigma^4$.

A lot of the statistical literature works with the definition of excess kurtosis; therefore subtracting 3 from (3.228) results in

$$\gamma_2 = \beta_2 - 3 = \frac{32 - 3\pi^2}{(4-\pi)^2} - 3 = -\frac{16 - 24\pi + 6\pi^2}{(4-\pi)^2}. \tag{3.229}$$

As with the coefficient of skewness for the Rayleigh distribution, we should note that (3.229) is also independent of the scale and shape parameters. Therefore, the coefficient of kurtosis for all Rayleigh distributions is approximately $\gamma_2 = 0.245$, another constant.

3.15.5 Median of the Rayleigh Distribution

Unlike with the gamma and beta distributions, it is possible to analytically evaluate the integral for the median state of the Rayleigh distribution. Thus, we have to solve

$$\int_0^{x_{med}} \frac{x}{\sigma^2} \exp\left\{-\frac{x^2}{2\sigma^2}\right\} dx = \frac{1}{2}. \tag{3.230}$$

The integral on the left-hand side of the equation in (3.230) integrates such that we can write (3.230) as

$$\begin{aligned}
\left[-\exp\left\{-\frac{x^2}{2\sigma^2}\right\}\right]_0^{x_{med}} &= \frac{1}{2}, \\
\Rightarrow 1 - \exp\left\{-\frac{x_{med}^2}{2\sigma^2}\right\} &= \frac{1}{2}, \\
\Rightarrow \exp\left\{-\frac{x_{med}^2}{2\sigma^2}\right\} &= \frac{1}{2}, \\
\Rightarrow \frac{x_{med}}{2\sigma^2} &= \ln 2, \\
\Rightarrow x_{med} &= \sigma\sqrt{2\ln 2}.
\end{aligned} \tag{3.231}$$

3.15.6 Mode of the Rayleigh Distribution

Unlike the many applications of integration by parts to find the different order non-central moments of the Rayleigh distribution, the derivation to find the mode of this distribution is quite simple. As with the other distributions presented so far, we are seeking the state, x_{mode}, such that the first derivative of the Rayleigh probability density function, evaluated at x_{mode}, is equal to zero. Therefore

$$\begin{aligned} x_{mode} \quad \text{s.t.} \quad & \left.\frac{dRa\left(\sigma^2\right)}{dx}\right|_{x=x_{mode}} = 0, \\ \Rightarrow & \frac{1}{\sigma^2}\exp\left\{-\frac{x^2}{2\sigma^2}\right\} - \frac{x^2}{\sigma^4}\exp\left\{-\frac{x^2}{2\sigma^2}\right\} = 0, \\ \Rightarrow & \frac{1}{\sigma^2} - \frac{x^2}{\sigma^4} \Rightarrow x_{mode} = \sigma. \end{aligned} \tag{3.232}$$

In summary, for the Rayleigh distribution we have that three descriptive statistics satisfy the inequality for a left-skewed distribution, mode $\leq$ median $\leq$ mean, which is

$$\sigma < \sigma\sqrt{2\ln 2} < \sigma\sqrt{\frac{\pi}{2}}.$$

Relationship of the Rayleigh distributions to other distributions

- If we have a random variable R that is distributed $Ra\left(\sigma\right)$, then the sum of the square of a set of N Rayleigh-distributed random variables, R_i is a gamma-distributed random variable with parameters N and $2\sigma^2$, i.e.,

$$Y = \sum_{i=1}^{N} R_i^2 \sim Ga\left(N, 2\sigma^2\right).$$

- The Rayleigh distribution is related to the Gaussian distribution through the property that if we have two independent normally distributed random variables $X \sim N\left(0, \sigma^2\right)$ and $Y \sim N\left(0, \sigma^2\right)$, then the random variable $R = \sqrt{X^2 + Y^2}$ is a Rayleigh-distributed random variable with parameter σ.
- If R is a Rayleigh-distributed random variable with $\sigma = 1$, then the random variable $Q = R^2$ has a χ^2 distribution with $N = 2$ degrees of freedom.
- If X is an exponentially distributed random variable such that $X \sim Ex\left(\lambda\right)$, then the transformed random variable $Y \equiv \sqrt{2X\sigma^2\lambda}$ is a Rayleigh-distributed random variable, $Y \sim Ra\left(\sigma\right)$.

3.16 Weibull Distribution

The Weibull distribution is named after the first person to define and use it, Waloddi Weibull, a Swedish physicist. Weibull used the distribution to model the distribution of breaking strength of materials. A problem with the Weibull distribution is the justification from a probabilistic point of view for its use; however, there is a lot of research into this justification. The definition of the Weibull distribution is as follows.

Definition 3.51. A random variable X is said to be a Weibull-distributed continuous random variable if there are values of the parameters, c and α, greater than zero, and ξ_0, such that

$$Y = \left(\frac{X - \xi_0}{\alpha}\right)^c$$

has the standard exponential distribution with probability density function

$$Exp\,(y) = \exp\{-y\}, \quad y > 0,$$

then the probability density function of the Weibull distribution for the continuous random variable, X, is

$$We\,(\alpha, c, \xi_0) = \frac{c}{\alpha}\left(\frac{x - \xi_0}{\alpha}\right)^{c-1} \exp\left\{\left(\frac{x - \xi_0}{\alpha}\right)^c\right\}, \quad x > \xi_0. \tag{3.233}$$

As with the previous distributions in this chapter, a standard version of the Weibull distribution exists which is defined when $\alpha = 1$ and $\xi_0 = 0$, therefore the standard Weibull distribution is given by

$$We\,(1, c, 0) \equiv cx^{c-1} \exp\left\{-x^c\right\}, \quad x > 0, c > 0. \tag{3.234}$$

In Fig. 3.18 we have plotted four different versions of the Weibull distribution, where we have kept $\alpha = 1$ and taken different values for c. We can see that as the parameter c increases, the distribution's

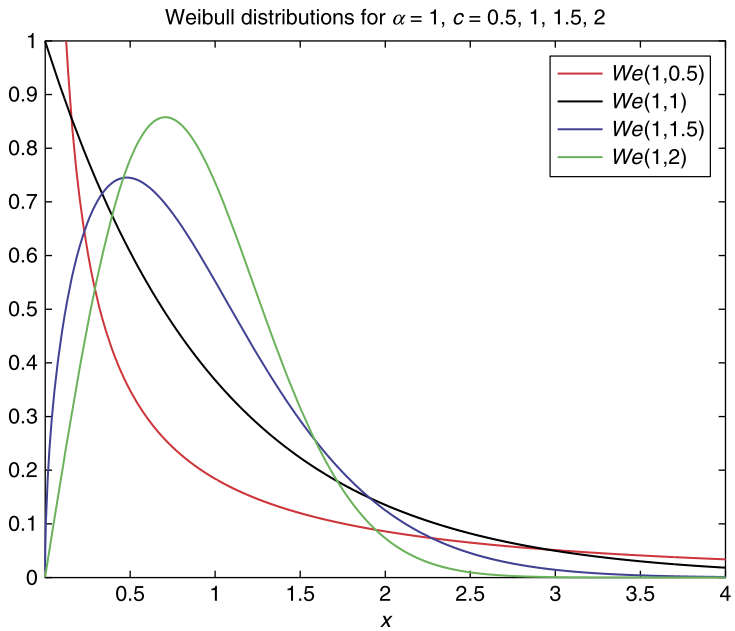


FIGURE 3.18

Plots of different versions of the Weibull distribution.

mode shifts to the right. Again the distribution appears similar to a lognormal in that they are left-skewed, but again the steepness of the function to the left of the mode does not appear to be as prevalent as for the lognormal distribution.

3.16.1 Moments of the Weibull Distribution

The derivation of the non-central moments of the Weibull is not straightforward, and so we shall work through step by step to obtain their expressions. We start with the mean, but to make things easier we assume that $\xi_0 = 0$. Therefore, the starting point for the mean of the Weibull distribution is

$$\mu = \mathbb{E}[X] \equiv \int_0^\infty x \frac{c}{\alpha^c} x^{c-1} \exp\left\{-\left(\frac{x}{\alpha}\right)^c\right\} dx. \tag{3.235}$$

The expression above does look quite daunting to integrate, but there are a couple of change of variables that will make it simpler. The first is to introduce the new constant, $\beta = \frac{1}{\alpha^c}$. This enables us to write (3.235) as

$$\mu = \int_0^\infty c\beta x^{c-1} \exp\left\{-\beta x^c\right\} dx.$$

The actual change of variable occurs through considering $t = \beta x^c$, which implies that x is equal to

$$\begin{aligned} t = \beta x^c &\Rightarrow \ln t = \ln\beta + c\ln x, \Rightarrow \ln t - \ln\beta = c\ln x, \\ &\Rightarrow \ln\frac{t}{\beta} = c\ln x, \Rightarrow \frac{1}{c}\ln\frac{t}{\beta} = \ln x, \\ &\Rightarrow \ln\left(\left(\frac{t}{\beta}\right)^{\frac{1}{c}}\right) = \ln x, \\ &\Rightarrow x = \left(\frac{t}{\beta}\right)^{\frac{1}{c}}. \end{aligned} \tag{3.236}$$

The next step is to determine the integrand with respect to t, which is $dt = c\beta x^{c-1} dx$. We then need to verify the limits of integration to be $x = 0 \Rightarrow t = 0$ and $x = \infty \Rightarrow t = \infty$. Therefore, it is possible to rewrite (3.235) in terms of t as

$$\mu = \int_0^\infty \left(\frac{t}{\beta}\right)^{\frac{1}{c}} \exp\{-t\} dt = \left(\frac{1}{\beta}\right)^{\frac{1}{c}} \int_0^\infty t^{\frac{1}{c}} \exp\{-t\} dt = \left(\frac{1}{\beta}\right)^{\frac{1}{c}} \int_0^\infty t^{\frac{1}{c}+1-1} \exp\{-t\} dt. \tag{3.237}$$

The important feature to note about the last term in (3.237) is that by using the regular mathematical trick of $-1 + 1 = 0$, it is possible to separate out the $+1$ term, which then makes the last term equivalent to a gamma function. Therefore, it is possible to rewrite (3.237) as

$$\mu = \left(\frac{1}{\beta}\right)^{\frac{1}{c}} \Gamma\left(1 + \frac{1}{c}\right) \equiv \alpha\Gamma\left(\frac{1+c}{c}\right).$$

To derive the expression for the variance of the Weibull distribution, we require $\mathbb{E}\left[X^2\right]$, which in turn requires the evaluation of

$$\mu_2' = \mathbb{E}\left[X^2\right] \equiv \int_0^\infty x^2 \frac{c}{\alpha^c} x^{c-1} \exp\left\{-\left(\frac{x}{\alpha}\right)^c\right\} dx. \tag{3.238}$$

Introducing the same change of variable for the mean enables (3.238) to be written as

$$\mu_2' = \mathbb{E}\left[X^2\right] \equiv \int_0^\infty \left(\frac{t}{\beta}\right)^{\frac{2}{c}} \exp\{-t\}\, dt. \tag{3.239}$$

Applying the same add one and then subtract one trick to the power in (3.239), and separating out the term that is not a function t, enables (3.239) to be written in terms of a gamma function as

$$\mu_2' = \left(\frac{1}{\beta}\right)^{\frac{2}{c}} \Gamma\left(1 + \frac{2}{c}\right) \equiv \alpha^2 \Gamma\left(\frac{2+c}{c}\right). \tag{3.240}$$

Therefore, it is possible to define the variance for the Weibull distribution as

$$Var\left[X\right] \equiv \mathbb{E}\left[X^2\right] - \left(\mathbb{E}\left[X\right]\right)^2 = \alpha^2 \left(\Gamma\left(\frac{2+c}{c}\right) - \Gamma\left(\frac{1+c}{c}\right)^2\right). \tag{3.241}$$

3.16.2 Skewness and Kurtosis of the Weibull Distribution

Given the expression for the mean and the non-central second moment, it is possible to notice that there is a general pattern to the expression of the associated gamma function. Therefore, we can show that the nth order non-central moment for the Weibull distribution can be written as

$$\mu_n' \equiv \mathbb{E}\left[X^n\right] = \alpha^n \Gamma\left(1 + \frac{n}{c}\right). \tag{3.242}$$

Thus, given the formula in (3.242), it is easy to show that the coefficient of skewness for the Weibull distribution is

$$\beta_1 = \frac{\Gamma\left(1+\frac{3}{c}\right) - 3\Gamma\left(1+\frac{2}{c}\right)\Gamma\left(1+\frac{1}{c}\right) + 2\Gamma\left(1+\frac{1}{c}\right)^3}{\Gamma\left(1+\frac{2}{c}\right)\sqrt{\Gamma\left(1+\frac{2}{c}\right)}}. \tag{3.243}$$

Note: The skewness coefficient for the Weibull distribution is only a function of the shape parameter c, as the α^3 terms cancel each other out.

Finally we consider the kurtosis of the Weibull distribution, which can easily be shown to be

$$\gamma_2 = \frac{\Gamma_4 - 4\Gamma_3\Gamma_1 + 6\Gamma_2\Gamma_1^2 - 3\Gamma_1^4}{\left(\Gamma_2 - \Gamma_1^2\right)^2}, \tag{3.244}$$

where

$$\Gamma_i \equiv \Gamma\left(1 + \frac{i}{c}\right).$$

Median of the Weibull distribution

It is possible to evaluate the integral expression for the median for the Weibull distribution. Therefore, the median x_{med} of the Weibull distribution is the state such that

$$x_{med} \quad \text{s.t.} \quad \int_0^{x_{med}} \frac{c}{\alpha^c} x^{c-1} \exp\left\{-\left(\frac{x}{\alpha}\right)^{\alpha}\right\} dx = \frac{1}{2}. \tag{3.245}$$

The integral in (3.245) can be evaluated analytically as

$$\begin{aligned}
\left[-\exp\left\{-\left(\frac{x}{\alpha}\right)^{c}\right\}\right]_0^{x_{med}} &= \frac{1}{2}, \\
\Rightarrow -\exp\left\{-\left(\frac{x_{med}}{\alpha}\right)^{c}\right\} + 1 &= \frac{1}{2}, \\
\Rightarrow \exp\left\{-\left(\frac{x_{med}}{\alpha}\right)^{c}\right\} &= \frac{1}{2}, \\
\Rightarrow -\left(\frac{x_{med}}{\alpha}\right)^{c} &= -\ln 2, \\
\Rightarrow x_{med} &= \alpha\,(\ln 2)^{\frac{1}{c}}.
\end{aligned} \tag{3.246}$$

3.16.3 Mode of the Weibull Distribution

Moving on to the mode, we again require the value for x such that the first derivative of (3.233) evaluated at this point is equal to zero. Therefore, we have

$$\begin{aligned}
&\frac{dWe(\alpha, c)}{dx} = 0, \\
&\Rightarrow (c-1)\,x^{c-2} \exp\left\{-\left(\frac{x}{\alpha}\right)^{c}\right\} - x^{c-1}\left(\frac{cx^{c-1}}{\alpha^c}\right) \exp\left\{-\left(\frac{x}{\alpha}\right)^{c}\right\} = 0, \\
&\Rightarrow x^{c-2} \exp\left\{-\left(-\frac{x}{\alpha}\right)^{c}\right\}\left((c-1) - x^c \frac{c}{\alpha^c}\right) = 0.
\end{aligned}$$

After some rearranging, it is possible to show that the mode of the Weibull distribution is

$$x_{mode} = \alpha\left(\frac{c-1}{c}\right)^{\frac{1}{c}}. \tag{3.247}$$

There is one condition for the mode of the Weibull distribution to exist: $c > 1$. If $c = 1$, then the mode is not defined. If $c < 1$, then the value for the mode is a negative number which cannot be for the Weibull distribution, as it is only defined for $x \geq 0$.

3.17 Gumbel Distribution

In probability theory and statistics, the Gumbel distribution is used to model the distribution of the maximum (or minimum) of a number of samples of various distributions. In hydrology, this distribution can be used to represent the distribution of the maximum level of a river in a particular year if there

was a list of maximum values for the past ten years. It is useful in predicting the chance that an extreme earthquake, flood, or other natural disaster will occur. The distribution was derived by Emil Julius Gumbel (1891–1966). The distribution is considered part of the extreme value family of distributions. We shall not go into that theory here, but present this distribution and its properties because of its use in geosciences to demonstrate this distribution.

We shall denote a generalized Gumbel distribution by $Gu\,(\alpha, \beta)$, where α is the scale parameter and β is the shape parameter. Therefore, the generalized version of the Gumbel distribution is given by

$$Gu\,(\alpha, \beta) \equiv \frac{1}{\beta} \exp\left\{\left(-\frac{x-\alpha}{\beta}\right) - \exp\left\{\left(-\frac{x-\alpha}{\beta}\right)\right\}\right\}, \quad -\infty < x < \infty. \tag{3.248}$$

The standardized version of the Gumbel distribution, $Gu\,(0, 1)$, is defined as

$$Gu\,(0, 1) = \exp\{-x - \exp\{-x\}\} \quad -\infty < x < \infty. \tag{3.249}$$

In Fig. 3.19, we have plotted different versions of the Gumbel distribution to illustrate the shape of the distribution. The first striking feature to note about the Gumbel distribution is that it is right-skewed; all but the beta distribution were left-skewed or not skewed at all. We can see that the effects the parameters have on the distribution differ quite a bit. The first three Gumbel distributions plotted remain close together as they have the same variance and the mode is simply shifting with the change in α. When we consider different combinations of the parameters α and β, we see quite different

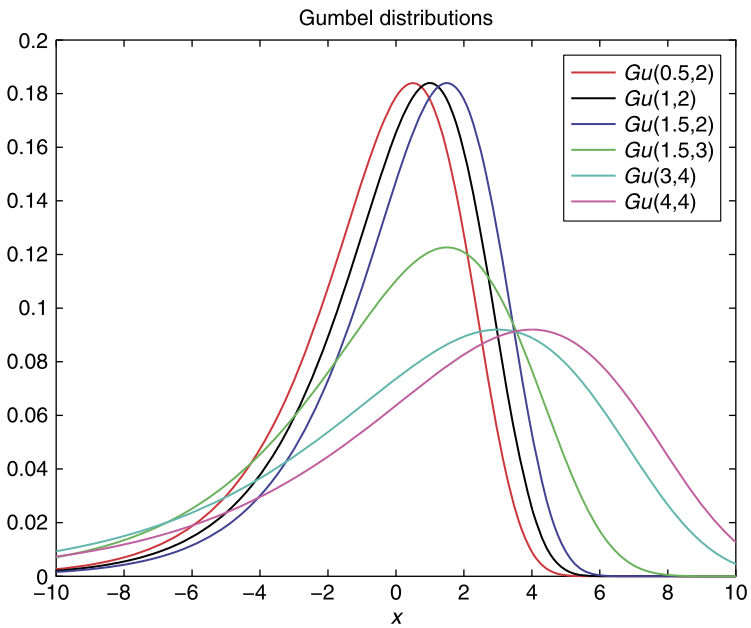


FIGURE 3.19

Plots of different versions of the Gumbel distribution.

responses, but we must remark that none of the distributions plotted in Fig. 3.19 resembles any of the other distributions we have so far considered in this chapter.

3.17.1 Moments of the Gumbel Distribution

While it may again look quite daunting to derive the MGF for the Gumbel distribution in (3.248), it is not as complicated as it may appear. The definition for the MGF for the Gumbel distribution is

$$\mathbb{E}\left[\exp\{tx\}\right] \equiv \int_{-\infty}^{\infty} \frac{1}{\beta} \exp\{tx\} \exp\left\{\left(-\frac{x-\alpha}{\beta}\right) - \exp\left\{\left(-\frac{x-\alpha}{\beta}\right)\right\}\right\} dx. \tag{3.250}$$

We now introduce the change of variable $u \equiv \exp\left\{-\frac{x-\alpha}{\beta}\right\}$, which can be rearranged to find an expression for x in terms of u as $x = \alpha - \beta \ln u$. Checking the limits of integration, when $x = -\infty$ then $u = \infty$, and when $x = \infty$ then $u = 0$. Therefore, the limits of integration are for $u \in [\infty, 0)$. The final expression to find is the integrand, which is $du = -\frac{1}{\beta} \exp\left\{-\frac{x-\alpha}{\beta}\right\} dx$. The minus sign that arises for changing the integrand enables us to switch the integration limits around. Therefore, given all this information, we can now rewrite (3.250) in terms of u as

$$\begin{aligned}
\mathbb{E}\left[\exp\{tx\}\right] &\equiv \int_{0}^{\infty} \exp\{\alpha t - \beta t \ln u\} \exp\{-u\}\, du, \\
&= \exp\{\alpha t\} \int_{0}^{\infty} u^{-\beta t} \exp\{-u\}\, du, \\
&= \exp\{\alpha t\}\, \Gamma(1 - \beta t).
\end{aligned} \tag{3.251}$$

To find the mean of the Gumbel distribution, we need to differentiate the MGF in (3.251) with respect to t and then set $t = 0$. This results in

$$\mu = \frac{d e^{\alpha t} \Gamma(1-\beta t)}{dx}\bigg|_{t=0} = \left(\alpha e^{\alpha t} \Gamma(1-\beta t) + e^{\alpha t} \frac{d}{dx} \Gamma(1-\beta t)\right)\bigg|_{t=0}. \tag{3.252}$$

We have not discussed the differentiation of the gamma function before, but it plays an important part in the derivation of the non-central moments of the Gumbel distribution. Therefore, differentiating (3.252) results in

$$\mu = \alpha \Gamma(1) - \beta \Gamma'(1) = \alpha + \gamma\beta, \tag{3.253}$$

where the constant, $\gamma = -\Gamma'(1)$, is referred to as the **Euler**, or the **Euler-Mascheroni**, constant and is approximately equal to 0.57721.

3.17.2 Differentiating Gamma Functions

However, before we can move on to the derivation of the variance, we need to derive some series approximations to various derivatives of the gamma function. We start with the first derivative and define the function, $\Psi(x)$, as

$$\Psi(x) \equiv \frac{d}{dx} \ln \Gamma(x) = \frac{\Gamma'(x)}{\Gamma(x)}. \tag{3.254}$$

The next step is to consider the formula $\Gamma(x+1) = x\Gamma(x)$, such that differentiating the formula with respect to x enables us to obtain properties of $\Psi(x)$, therefore,

$$\begin{aligned}
\Gamma'(x+1) &= \Gamma(x) + x\Gamma'(x), \\
\frac{\Gamma(x+1)}{\Gamma(x)} &= 1 + x\frac{\Gamma'(x)}{\Gamma(x)}, \\
x\frac{\Gamma'(x)}{\Gamma(x)} &= 1 + x\frac{\Gamma'(x)}{\Gamma(x)}, \\
\Psi(x+1) &= \frac{1}{x} + \Psi(x),
\end{aligned} \tag{3.255}$$

where we have substituted $\Gamma(x+1) = x\Gamma(x)$ on the left-hand side in (3.255), and assume that $x \neq 0$, specifically that $x > 0$.

If we consider the case for $x \in \mathbb{N}^+$, then it is possible to expand (3.255) as

$$\Psi(n+1) = \frac{1}{n} + \Psi(n) = \frac{1}{n} + \frac{1}{n-1} + \Psi(n-1) = \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \Psi(n-2) = \cdots.$$

Continuing the expansion until we arrive at $n = 1$, which we already know is $-\gamma$, results in a series expansion for $\Psi(n+1)$ as

$$\Psi(n+1) = -\gamma + \sum_{k=1}^{n} \frac{1}{k}, \quad n = 0, 1, 2, \ldots. \tag{3.256}$$

Before we can continue, we have to derive the formulae for Stirling's approximation. We have briefly seen this formula in the derivation of the Gaussian distribution from the binomial distribution. The basis of Stirling's formula is to consider the logarithm of n-factorial, which is

$$\begin{aligned}
\ln n! &= \ln 1.2.3 \ldots .n = \ln 1 + \ln 2 + \ln 3 + \cdots + \ln n, \\
&= \sum_{k=1}^{n} \ln k, \\
&\approx \int_1^n \ln x dx = [x \ln x]_1^n = n \ln n - n + 1 \approx n \ln n - n.
\end{aligned}$$

As we saw from the definition of the gamma function for integers, it is possible to write $n!$ as

$$n! = \int_0^\infty e^{-x} x^n dx. \tag{3.257}$$

Next, if we differentiate the logarithm of the integrand above, we have

$$\frac{d}{dx} \ln\left(e^{-x} x^n\right) = \frac{d}{dx}(n \ln x - x) = \frac{n}{x} - 1. \tag{3.258}$$

The integrand above is sharply peaked around $x = n$, therefore let $x = n + \xi$, where it is assumed that ξ is much smaller than n. This enables us to write (3.258) as

$$\ln\left(x^n e^{-x}\right) = n \ln(n+\xi) - (n+\xi). \tag{3.259}$$

It is possible to write

$$\begin{aligned}\ln(n+\xi) &\equiv \ln n\left(1+\frac{\xi}{n}\right),\\ &= \ln n + \ln\left(1+\frac{\xi}{n}\right),\\ &= \ln n + \frac{\xi}{n} - \frac{1}{2}\frac{\xi^2}{n^2} + \cdots,\end{aligned} \tag{3.260}$$

where we have used the Taylor series for $\ln(1+x)$.

Substituting (3.260) into (3.259) yields

$$\begin{aligned}\ln\left(x^n e^{-x}\right) &= n\ln(n+\xi) - (n+\xi),\\ &= n\ln n + \xi - \frac{1}{2}\frac{\xi^2}{n} - n - \xi + \cdots,\\ &= n\ln n - n - \frac{\xi^2}{2n} + \cdots.\end{aligned} \tag{3.261}$$

Taking the exponential of (3.261) results in

$$x^n e^{-x} = e^{n\ln n} e^{-n} e^{-\frac{\xi^2}{2n}} = n^n e^{-n} e^{-\frac{\xi^2}{2n}}. \tag{3.262}$$

Substituting (3.262) into (3.257) yields

$$n! \approx \int_{-n}^{\infty} n^n e^{-n} e^{-\frac{\xi^2}{2n}} dx \approx n^n e^{-n} \int_{-\infty}^{\infty} e^{-\frac{\xi}{2n}} dx. \tag{3.263}$$

An important feature to notice here is that the integral in (3.263) is that of $\sqrt{2\pi n}$ times the cumulative density function of the normal distribution, which is equal to one. Therefore, we can write (3.263) as

$$n! \approx n^n e^{-n}\sqrt{2\pi n} = n^n + \frac{1}{2} e^{-n}\sqrt{2\pi}. \tag{3.264}$$

Taking the logarithm of (3.264) results in Stirling's formula

$$\ln\Gamma(n+1) = \ln n! \approx \left(x+\frac{1}{2}\right)\ln x - x + \frac{1}{2}\ln 2\pi + O\left(x^{-1}\right), \quad \text{as } n\to\infty. \tag{3.265}$$

The results that have been derived above are a form of an asymptotic expansion of $\Psi(n)$ as $n\to\infty$. Differentiating (3.265) yields the asymptotic behavior for large n of $\Psi(n+1)$, which is

$$\Psi(n+1) = \ln n + \frac{1}{2x} + O\left(x^{-2}\right), \quad \text{as } n\to\infty. \tag{3.266}$$

If we now consider the limit as $n\to\infty$ of $\ln\Psi(n+1) - \ln n$, we see from (3.266) that this is zero. Therefore, substituting (3.255) into (3.266) with this new limit information enables us to arrive to at an approximation to the Euler constant γ as

$$\gamma = \lim_{n\to\infty}\left(\sum_{k=1}^{n}\frac{1}{k} - \ln n\right). \tag{3.267}$$

Rearranging (3.267) and removing the limit, but including the recognition of the higher-order terms, we have

$$\sum_{k=1}^{n} \frac{1}{k} = \ln n + \gamma + \frac{1}{2n} + O\left(\frac{1}{n^2}\right). \tag{3.268}$$

However, so far the derivation has been for when x is a positive integer. We now consider the case for when x is arbitrary and extend the results proven above. Consider the term

$$\Psi(x+n) = \frac{1}{x+n-1} + \frac{1}{x+n-2} + \cdots + \frac{1}{x} + \Psi(x) \equiv \sum_{k=0}^{n-1} \frac{1}{x+k} + \Psi(x), \tag{3.269}$$

where we have substituted $x+n$ for n in (3.255) and $n \in \mathbb{N}^+$. Therefore, subtracting (3.256) from (3.269), we obtain

$$\Psi(x+n) - \Psi(n+1) = \sum_{k=0}^{n-1} \left(\frac{1}{x+k} - \frac{1}{k+1}\right) + \gamma + \Psi(x). \tag{3.270}$$

If we now consider the limit of (3.270) as $n \to \infty$, and substitute (3.266), we see that

$$\lim_{n\to\infty} \Psi(x+n) - \Psi(n+1) = O\left(n^{-1}\right) \to 0.$$

Therefore, given this information, it is now possible to write $\Psi(x)$ as

$$\Psi(x) = -\gamma - \sum_{k=0}^{\infty} \left(\frac{1}{x+k} - \frac{1}{k+1}\right), \tag{3.271}$$

$$\Rightarrow \frac{\Gamma'(x)}{\Gamma(x)} = -\gamma - \sum_{k=0}^{\infty} \left(\frac{1}{x+k} - \frac{1}{k+1}\right), \tag{3.272}$$

which now gives us a differentiable expression for the derivatives of the gamma functions.

3.17.3 Returning to the Moments of the Gumbel Distribution

The variance of any distribution requires the second derivative of its MGF, if it exists; therefore for the Gumbel distribution we require the second derivative of the gamma function $\Gamma(1-\beta t)$, which is

$$\frac{d^2 \ln \Gamma(1-\beta t)}{dt^2} = -\beta \frac{d\Psi(1-\beta t)}{dt} = \beta^2 \sum_{k=0}^{\infty} \frac{1}{(1-\beta t+k)^2}. \tag{3.273}$$

It is possible to find the second derivative of the gamma functions as

$$\frac{d^2 \ln \Gamma(1-\beta t)}{dt^2} = \frac{d}{dx}\frac{\Gamma'(1-\beta t)}{\Gamma(1-\beta t)} = \beta^2 \frac{\Gamma''(1-\beta t)}{\Gamma(1-\beta t)} - \left(\frac{\Gamma'(1-\beta t)}{\Gamma(1-\beta t)}\right)^2. \tag{3.274}$$

Setting $t = 0$ in both (3.273) and (3.274), we have that $\Gamma(1) = 1$ and $\Gamma'(1) = \beta\gamma$. Therefore, (3.274) becomes

$$\beta^2\Gamma''(1) - \beta^2\gamma^2 = \beta^2 \sum_{k=0}^{\infty} \frac{1}{(1+k)^2} = \beta^2 \frac{\pi}{6} \Rightarrow \beta^2\Gamma''(1) = \beta^2\gamma^2 + \beta^2 \frac{\pi^2}{6}. \tag{3.275}$$

Given an expression for the second derivative of $\Gamma(1-\beta t)$ evaluated at $t = 0$, we return to the definition of the variance which requires $\frac{d^2}{dx^2} e^{\alpha t}\Gamma(1-\beta t)$, which is

$$\begin{aligned} Var[X] &= \frac{d^2}{dx^2} e^{\alpha t}\Gamma(1-\beta t) - \left(\frac{d}{dx} e^{\alpha t}\Gamma(1-\beta t)\right)^2 \Big|_{t=0}, \\ &= \alpha^2 e^{\alpha t}\Gamma(1-\beta t) - 2\alpha\beta e^{\alpha t}\Gamma'(1-\beta t) + \beta^2 e^{\alpha t}\Gamma''(1-\beta t) \\ &\quad - \left(\alpha e^{\alpha t}\Gamma(1-\beta t)\right)^2 - 2\beta\alpha e^{\alpha t}\Gamma(1-\beta t)\Gamma'(1-\beta t) - \beta^2\Gamma'(1-\beta t)^2 \Big|_{t=0}, \\ &= \alpha^2 - 2\alpha\beta\gamma + \beta^2\left(\gamma^2 + \frac{\pi}{6}\right) - \alpha^2 - 2\alpha\beta\gamma - \beta^2\gamma^2, \\ &= \frac{\beta\pi}{6}. \end{aligned} \tag{3.276}$$

3.17.4 Skewness of a Gumbel Distribution

We now move on to the skewness of the Gumbel distribution, which is going to involve the differentiation of the gamma function in the MGF for this distribution. There are three terms that we have to consider in the numerator of the definition of skewness: $2(\mathbb{E}[X])^3$, $-3\mathbb{E}\left[X^2\right]\mathbb{E}[X]$, and $\mathbb{E}\left[X^3\right]$. Evaluating the $2(\mathbb{E}[X])^3$ term first, we have

$$\begin{aligned} 2\mathbb{E}[X]^3 &= 2\left(\frac{d}{dt}\left[e^{\alpha t}\Gamma(1-\beta t)\right]\right)\Big|_{t=0}^3, \\ &= 2(\alpha+\beta\gamma)^3, \\ &= 2\alpha^3 + 6\alpha^2\beta\gamma + 6\alpha\beta^2\gamma^2 + 2\beta^3\gamma^3. \end{aligned}$$

Moving on to the second term in the list above, we have

$$\begin{aligned} -3\mathbb{E}\left[X^2\right]E[X] &= -3\,\frac{d^2}{dt^2}\left(e^{\alpha t}\Gamma(1-\beta t)\right)\Big|_{t=0} \left(\frac{d}{dt} e^{\alpha t}\Gamma(1-\beta t)\right)\Big|_{t=0}, \\ &= \left(\alpha^2 + 2\alpha\beta\gamma + \beta^2\gamma^2 + \beta^2\frac{\pi^2}{6}\right)(\alpha+\beta\gamma), \\ &= -3\alpha^3 - 9\alpha^2\beta\gamma - 9\alpha\beta^2\gamma^2 - 3\alpha\beta^2\frac{\pi^2}{6} - 3\beta^3\gamma\frac{\pi^2}{6} - 3\beta^3\gamma^3. \end{aligned}$$

Combining the two terms above to keep track of all the terms involved so far, we have

$$\begin{array}{ccccccc} 2\alpha^3 & 6\alpha^2\beta\gamma & 6\alpha\beta^2\gamma^2 & 2\beta^3\gamma^3 & & \\ -3\alpha^3 & -9\alpha^2\beta\gamma & -9\alpha\beta^2\gamma^2 & -3\beta^3\gamma^3 & -3\alpha\beta^2\frac{\pi^2}{6} & -3\beta^3\gamma\frac{\pi^2}{6} \end{array} \tag{3.277}$$

We now move on to the last term, which involves a new term that we will have to derive. To find $\mathbb{E}\left[X^3\right]$, we require the third derivative of the moment-generating function, which is found by taking the derivative of the second derivative of the MGF. Thus we have

$$\frac{d}{dt}\left(\alpha^2 e^{\alpha t}\Gamma(1-\beta t) - 2\alpha\beta e^{\alpha t}\Gamma'(1-\beta t) + \beta^2 e^{\alpha t}\Gamma''(1-\beta t)\right)\Big|_{t=0},$$

which can be shown to be

$$\begin{aligned}\mathbb{E}\left[X^3\right] &= \alpha^3\Gamma - 3\alpha^2\beta\Gamma' + 3\beta^2\alpha\Gamma'' - \beta^3\Gamma''' \\ &= \alpha^3 + 3\alpha^2\beta\gamma + 3\alpha\beta^2\gamma^2 + 3\alpha\beta^2\frac{\pi^2}{6} - \beta^3\Gamma'''. \end{aligned} \tag{3.278}$$

Combining all of the α, β, and γ terms from (3.278) with (3.277), we have

$$\begin{array}{ccccccc} 2\alpha^3 & 6\alpha^2\beta\gamma & 6\alpha\beta^2\gamma^2 & 2\beta^3\gamma^3 & & & \\ -3\alpha^3 & -9\alpha^2\beta\gamma & -9\alpha\beta^2\gamma^2 & -3\beta^3\gamma^3 & -3\alpha\beta^2\frac{\pi^2}{6} & -3\beta^3\gamma\frac{\pi^2}{6} & . \\ \alpha^3 & 3\alpha^2\beta\gamma & 3\alpha\beta^2\gamma^2 & & 3\alpha\beta^2\frac{\pi^2}{6} & & \end{array} \tag{3.279}$$

The next step is to consider the third derivative of the gamma function, which requires us to evaluate

$$\frac{d^3}{dt^3}\ln\Gamma(1-\beta t) = \beta^2\frac{d}{dt}\left(\frac{\Gamma''(1-\beta t)}{\Gamma(1-\beta t)} - \frac{\Gamma'(1-\beta t)^2}{\Gamma(1-\beta t)^2}\right). \tag{3.280}$$

Applying both the chain rule and the product rule for differentiation to (3.280) results in

$$\frac{d^3}{dt^3}\ln\Gamma(1-\beta t) = -\beta^3\left(\frac{\Gamma'''(1-\beta t)}{\Gamma(1-\beta t)} - 3\frac{\Gamma''(1-\beta t)\Gamma'(1-\beta t)}{\Gamma(1-\beta t)^2} + 2\frac{\Gamma'(1-\beta t)^3}{\Gamma(1-\beta t)^3}\right). \tag{3.281}$$

Evaluating (3.281) at $t = 0$ results in

$$\begin{aligned} &-\beta^3\left(\Gamma'''(1) - 3\Gamma''(1)\Gamma'(1) + 2\left(\Gamma'(1)\right)^3\right), \\ &-\beta^3\Gamma'''(1) - 3\beta^3\gamma^3 - 3\beta^3\gamma\frac{\pi^2}{6} + 2\beta^3\gamma^3. \end{aligned} \tag{3.282}$$

However, we still have an expression in terms of $\Gamma'''(1)$. We recall that

$$\begin{aligned} \frac{d^2\ln\Gamma(1-\beta t)}{dt^2} &= \beta^2\sum_{k=0}^{\infty}\frac{1}{(1-\beta t+k)^2}, \\ \Rightarrow \frac{d^3\ln\Gamma(1-\beta t)}{dt^3} &= 2\beta^3\sum_{k=0}^{\infty}\frac{1}{(1-\beta t+k)^3}. \end{aligned} \tag{3.283}$$

Evaluating (3.283) at $t = 0$ results in the right-hand side being equal to $2\beta^3\zeta(3)$, where $\zeta(3)$ is **Apery's Constant**, which is a specific value for the **Riemann Zeta Function**. Therefore, substituting the expression above into (3.282) results in $\Gamma'''(1)$ being equivalent to

$$-\beta^3\Gamma'''(1) = 2\beta^3\zeta(3) + \beta^3\gamma^3 + 3\beta^3\gamma\frac{\pi^2}{6}. \tag{3.284}$$

Thus combining (3.284) with the information in (3.279) results in

$$\begin{array}{ccccccc} 2\alpha^3 & 6\alpha^2\beta\gamma & 6\alpha\beta^2\gamma^2 & 2\beta^3\gamma^3 & & & \\ -3\alpha^3 & -9\alpha^2\beta\gamma & -9\alpha\beta^2\gamma^2 & -3\beta^3\gamma^3 & -3\alpha\beta^2\frac{\pi^2}{6} & -3\beta^3\gamma\frac{\pi^2}{6} & \\ \alpha^3 & 3\alpha^2\beta\gamma & 3\alpha\beta^2\gamma^2 & \beta^3\gamma^3 & 3\alpha\beta^2\frac{\pi^2}{6} & 3\beta^3\gamma\frac{\pi^2}{6} & 2\beta^3\zeta\,(3) \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 2\beta^3\zeta\,(3)\,. \end{array} \tag{3.285}$$

Therefore, $\mathbb{E}\left[(X-\mathbb{E}\,(X))^3\right]=2\beta^3\zeta\,(3)$. The denominator of the definition for the coefficient of skewness for the Gumbel distribution can easily be verified to be $\frac{\pi^3}{6\sqrt{6}}$. Thus the coefficient of skewness for the Gumbel distribution is

$$\beta_1=\frac{12\sqrt{6}\zeta\,(3)}{\pi^3}, \tag{3.286}$$

which, after all the meticulous derivation above, is a **constant!**

3.17.5 Kurtosis of the Gumbel Distribution

The derivation of the kurtosis requires the uncentered fourth moment, which is found by identifying the fourth derivative of the moment-generating function evaluated at $t=0$. Applying the derivative operator to (3.278) results in

$$\begin{aligned}\mathbb{E}\left[X^4\right]=\Big(&\alpha^4e^{\alpha t}\Gamma\,(1-\beta t)-4\alpha^3\beta e^{\alpha t}\Gamma'\,(1-\beta t)+6\alpha^2\beta^2e^{\alpha t}\Gamma''\,(1-\beta t)\\&-4\alpha\beta^3e^{\alpha t}\Gamma'''\,(1-\beta t)+\beta^4e^{\alpha t}\Gamma''''\,(1-\beta t)\Big)\Big|_{t=0}.\end{aligned}$$

Forming an array, as we did with the skewness to keep track of the coefficients, we have

$$\alpha^4 \quad 4\alpha^3\beta\gamma \quad 6\alpha^2\beta^2\gamma^2 \quad \alpha^2\beta^2\pi^2 \quad 2\alpha\beta^3\gamma\pi^2 \quad 4\alpha\beta^3\gamma^3 \quad 8\alpha\beta^3\zeta\,(3) \quad \beta^4\Gamma''''\,(1)\,. \tag{3.287}$$

We shall deal with the fourth derivative of the gamma functions after we derive the other terms. The next term that we consider is $-4\mathbb{E}\left[X^3\right]\mathbb{E}\,[X]$, which is

$$\begin{aligned}&-4E\left[X^3\right]E\,[X]\\&=-4\left(\alpha^3+3\alpha^2\beta\gamma+3\alpha\beta^2\gamma^2+\beta^3\gamma^3+3\alpha\beta\frac{\pi^2}{6}+3\beta^3\gamma\frac{\pi^2}{6}+2\beta^3\zeta\,(3)\right)(\alpha+\beta\gamma)\,,\\&=-4\left(\alpha^4+4\alpha^3\beta\gamma+6\alpha^2\beta^2\gamma^2+4\alpha\beta^3\gamma^3+\alpha\beta^3\gamma\pi^2+3\alpha^2\beta^2\frac{\pi^2}{6}\right.\\&\quad\left.+3\beta^4\gamma^2\frac{\pi^2}{6}+2\beta^4\gamma\zeta\,(3)+2\alpha\beta^3\zeta\,(3)+\beta^4\gamma^4\right).\end{aligned}$$

Substituting this information into (3.287) requires the matrix to be expanded to

$$\begin{array}{ccccccc} \alpha^4 & 4\alpha^3\beta\gamma & 6\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 4\alpha\beta^3\gamma^3 & 8\alpha\beta^3\zeta(3) \quad \beta^4\Gamma''''(1) \\ -4\alpha^4 & -16\alpha^3\beta\gamma & -24\alpha^2\beta^2\gamma^2 & -2\alpha^2\beta^2\pi^2 & -4\alpha\beta^3\gamma\pi^2 & -16\alpha\beta^3\gamma^3 & -8\alpha\beta^3\zeta(3) \\ -4\beta^4\gamma^4 & -8\beta^4\gamma\zeta(3) & -2\beta^4\gamma^2\pi^2 & . & & & \end{array} \tag{3.288}$$

The next term to consider is $6\mathbb{E}\left[X^2\right]\mathbb{E}[X]^2$, which is equal to

$$\begin{aligned} 6\mathbb{E}\left[X^2\right]\mathbb{E}[X]^2 &= \left(\alpha^2+2\alpha\beta\gamma+\beta^2\gamma^2\right)\left(\alpha^2+2\alpha\beta\gamma+\beta^2\gamma^2+\beta^2\frac{\pi^2}{6}\right), \\ &= 6\alpha^4+24\alpha^3\beta\gamma+36\alpha^2\beta^2\gamma^2+6\alpha^2\beta^2\pi^2+24\alpha\beta^3\gamma^3+2\alpha\beta^3\gamma\pi^2+\beta^4\gamma^2\pi^2+6\beta^4\gamma^4. \end{aligned}$$

Collecting this information into (3.289) results in

$$\begin{array}{ccccccc} \alpha^4 & 4\alpha^3\beta\gamma & 6\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 4\alpha\beta^3\gamma^3 & 8\alpha\beta^3\zeta(3) \quad \beta^4\Gamma''''(1) \\ -4\alpha^4 & -16\alpha^3\beta\gamma & -24\alpha^2\beta^2\gamma^2 & -2\alpha^2\beta^2\pi^2 & -4\alpha\beta^3\gamma\pi^2 & -16\alpha\beta^3\gamma^3 & -8\alpha\beta^3\zeta(3) \\ 6\alpha^4 & 24\alpha^3\beta\gamma & 36\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 24\alpha\beta^3\gamma^3 & \\ -4\beta^4\gamma^4 & -8\beta^4\gamma\zeta(3) & -2\beta^4\gamma^2\pi^2 & & & & \\ 6\beta^4\gamma^4 & & \beta^4\gamma^2\pi^2 & . & & & \end{array} \tag{3.289}$$

The final term to consider is $-3\left(\mathbb{E}[X]\right)^4 \equiv -3\left(\alpha+\beta\gamma\right)^4$, which can easily be shown to be

$$-3\left(\mathbb{E}[X]\right)^4 = -3\alpha^3 - 12\alpha^3\beta\gamma - 18\alpha^2\beta^2\gamma^2 - 12\alpha\beta^3\gamma^3 - 3\beta^4\gamma^4.$$

Collecting this last information into (3.289) results in

$$\begin{array}{ccccccc} \alpha^4 & 4\alpha^3\beta\gamma & 6\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 4\alpha\beta^3\gamma^3 & 8\alpha\beta^3\zeta(3) \quad \beta^4\Gamma''''(1) \\ -4\alpha^4 & -16\alpha^3\beta\gamma & -24\alpha^2\beta^2\gamma^2 & -2\alpha^2\beta^2\pi^2 & -4\alpha\beta^3\gamma\pi^2 & -16\alpha\beta^3\gamma^3 & -8\alpha\beta^3\zeta(3) \\ 6\alpha^4 & 24\alpha^3\beta\gamma & 36\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 24\alpha\beta^3\gamma^3 & \\ -3\alpha^4 & -12\alpha^3\beta\gamma & -18\alpha^2\beta^2\gamma^2 & & & -12\alpha\beta^3\gamma^3 & \\ -4\beta^4\gamma^4 & -8\beta^4\gamma\zeta(3) & -2\beta^4\gamma^2\pi^2 & & & & \\ 6\beta^4\gamma^4 & & \beta^4\gamma^2\pi^2 & . & & & \\ -3\beta^4\gamma^4 & & & & & & \end{array} \tag{3.290}$$

Returning to consider the fourth derivative of the gamma function, we have

$$\begin{aligned} \Gamma''''(1-\beta t)\big|_{t=0} &= \frac{d}{dt}\left(\frac{\Gamma'''(1-\beta t)}{\Gamma(1-\beta t)} - 3\frac{\Gamma''(1-\beta t)\,\Gamma'(1-\beta t)}{\Gamma(1-\beta t)^2} + 2\frac{\left(\Gamma'(1-\beta t)\right)^3}{\Gamma(1-\beta t)^3}\right)\Bigg|_{t=0} \\ &= 6\beta^4\sum_{k=0}^{\infty}\frac{1}{(1-\beta t+k)}\Bigg|_{t=0} = 6\beta^4\zeta(4) = \frac{6\beta^4\pi^4}{90} = \frac{\beta^4\pi^4}{15}. \end{aligned}$$

Applying the derivative operator above results in

$$\begin{aligned}\Gamma''''(1) &= \frac{\Gamma''''(1)}{\Gamma(1)} - 4\frac{\Gamma'''(1)\Gamma'(1)}{\Gamma(1)^2} - 3\frac{\Gamma''(1)^2}{\Gamma(1)^2} + 12\frac{\Gamma'(1)^2\Gamma''(1)}{\Gamma(1)^3} - 6\frac{\Gamma'(1)^4}{\Gamma(1)^4} \\ &= \beta^4\frac{\pi^4}{15}.\end{aligned} \tag{3.291}$$

We have already derived expressions for all the derivatives of the gamma function up to the third order derivative. Substituting these expressions into (3.291) and rearranging to isolate the fourth derivative of the gamma function results in

$$\beta^4\Gamma''''(1) = 8\beta^4\Gamma\zeta(3) + \beta^4\gamma^4 + \beta^4\gamma^2\pi^2 + \frac{27\beta^4\pi^4}{180}. \tag{3.292}$$

Substituting the expressions in (3.292) into (3.290) in place of $\Gamma''''(1)$ results in

$$\begin{array}{rrrrrrr}
\alpha^4 & 4\alpha^3\beta\gamma & 6\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 4\alpha\beta^3\gamma^3 & 8\alpha\beta^3\zeta(3) \\
-4\alpha^4 & -16\alpha^3\beta\gamma & -24\alpha^2\beta^2\gamma^2 & -2\alpha^2\beta^2\pi^2 & -4\alpha\beta^3\gamma\pi^2 & -16\alpha\beta^3\gamma^3 & -8\alpha\beta^3\zeta(3) \\
6\alpha^4 & 24\alpha^3\beta\gamma & 36\alpha^2\beta^2\gamma^2 & \alpha^2\beta^2\pi^2 & 2\alpha\beta^3\gamma\pi^2 & 24\alpha\beta^3\gamma^3 & \\
-3\alpha^4 & -12\alpha^3\beta\gamma & -18\alpha^2\beta^2\gamma^2 & & & -12\alpha\beta^3\gamma^3 & \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

$$\begin{array}{rrrr}
\beta^4\Gamma^4 & 8\beta^4\gamma\zeta(3) & \beta^4\gamma^2\pi^2 & \beta^4\frac{27\pi^4}{180} \\
-4\beta^4\gamma^4 & -8\beta^4\gamma\zeta(3) & -2\beta^4\gamma^2\pi^2 & \\
6\beta^4\gamma^4 & & \beta^4\gamma^2\pi^2 & \\
-3\beta^4\gamma^4 & & & \\
\hline
0 & 0 & 0 & \beta^4\frac{27\pi^4}{180}
\end{array}, \tag{3.293}$$

which after all this work is again a constant

Finally, recalling that the definition of the kurtosis is the ratio of the fourth order non-central moment to the square of variance, the latter for the Gumbel distribution can easily be shown to be $\frac{\beta^4\pi^4}{36}$. Thus the kurtosis of the Gumbel distribution is another constant given by

$$\beta_2 = \frac{\frac{27\pi^4}{180}}{\frac{\pi^4}{36}} = \frac{27}{5}. \tag{3.294}$$

3.17.6 Median of the Gumbel Distribution

As with all of the other distributions that have been presented, when it is possible to evaluate the integral, the definition for the median is the value x_{med} such that

$$\int_{-\infty}^{x_{med}} \frac{1}{\beta}\exp\left\{\left(\frac{x-\alpha}{\beta}\right) - \exp\left\{\left(\frac{x-\alpha}{\beta}\right)\right\}\right\} dx = \frac{1}{2}. \tag{3.295}$$

The important feature to note here is that as we showed for the derivation of the mode, the derivative of $\exp\left\{-\exp\left\{-\frac{x-\alpha}{\beta}\right\}\right\}$ is $\frac{1}{\beta}\exp-\left\{\frac{x-\alpha}{\beta}\right\}\exp\left\{-\exp\left\{-\frac{x-\alpha}{\beta}\right\}\right\}$ is the definition of the Gumbel distribution, and thus it is possible to integrate (3.295). Therefore, (3.295) becomes

$$\left[\exp\left\{-\exp\left\{-\frac{x-\alpha}{\beta}\right\}\right\}\right]_{-\infty}^{x_{med}}=\frac{1}{2}. \tag{3.296}$$

The lower limit in (3.296) is equal to zero, which then leaves

$$\begin{aligned}\exp\left\{-\exp\left\{-\frac{x_{med}-\alpha}{\beta}\right\}\right\}&=\frac{1}{2},\\ \Rightarrow-\exp\left\{-\frac{x_{med}-\alpha}{\beta}\right\}&=-\ln 2,\\ \Rightarrow-\frac{x_{med}-\alpha}{\beta}&=\ln(\ln 2),\\ \Rightarrow x_{med}&=\alpha-\beta\ln(\ln 2).\end{aligned} \tag{3.297}$$

The constant $\ln(\ln 2)$ is approximately equal to 0.366573, so the median of the Gumbel distribution is approximately $x_{med}=\alpha-0.366573\beta$.

3.17.7 Mode of the Gumbel Distribution

As with all the other distributions, we require the state x_{mode} such that the first derivative of the Gumbel PDF, evaluated at this state, is zero. It is quite clear that the definition of Gumbel PDF in (3.248) could be quite difficult to evaluate, but it is not. The first tool that we need is the rules for differentiating a composite of functions. We can express (3.248) in terms of general functions $f(x)$ and $g(h(x))$, which implies that

$$\begin{aligned}\left.\frac{dGu(\alpha,\beta)}{dx}\right|_{x=x_{mode}}=0\Rightarrow\frac{d}{dx}\left(\frac{1}{\beta}\exp\{f(x)-g(h(x))\}\right)=0,\\ \Rightarrow\left(f'(x)-h'(x)g'(h(x))\right)\left(\frac{1}{\beta}\exp\{f(x)-g(h(x))\}\right)=0.\end{aligned}$$

Therefore, for the Gumbel distribution $f(x)=\frac{x-\alpha}{\beta}\Rightarrow f'(x)=\frac{1}{\beta}$ and $g(h(x))=\exp\left\{\frac{x-\alpha}{\beta}\right\}\Rightarrow h'(x)g'(h(x))=\frac{1}{\beta}\exp\left\{\frac{x-\alpha}{\beta}\right\}$. Given these expressions, we can write the first derivative of (3.248) as

$$\frac{1}{\beta}\left(\frac{1}{\beta}-\frac{1}{\beta}\exp\left\{\frac{x-\alpha}{\beta}\right\}\right)\exp\left\{\left(\frac{x-\alpha}{\beta}\right)-\exp\left\{\left(\frac{x-\alpha}{\beta}\right)\right\}\right\}=0. \tag{3.298}$$

The only way that the expression in (3.298) can be equal to zero is if the term multiplying the more complicated exponential term is zero. This can only occur when the exponential is equal to one, which means that $x-\alpha=0$. Therefore, the mode of the Gumbel distribution is $x_{mode}=\alpha$.

Table 3.4 A Summary of the Moment-Generating Functions and Different Standardized Moments for the Continuous Distributions Presented in This Chapter.

Distribution	MGF	Mean	Variance	Skewness	Kurtosis
Gaussian	$e^{\mu t+\frac{\sigma^2 t^2}{2}}$	μ	σ^2	0	3
Lognormal	None	$e^{\mu+\frac{\sigma^2}{2}}$	$\left(e^{\sigma^2}-1\right)e^{2\mu+\sigma^2}$	$\left(e^{\sigma^2}+2\right)\sqrt{\left(e^{\sigma^2}-1\right)}$	$e^{4\sigma^2}+2e^{3\sigma^2}+3e^{2\sigma^2}$
Reverse-lognormal	None	$t-e^{\mu+\frac{\sigma^2}{2}}$	$\left(e^{\sigma^2}-1\right)e^{2\mu+\sigma^2}$	$-\left(e^{\sigma^2}+2\right)\sqrt{\left(e^{\sigma^2}-1\right)}$	$e^{4\sigma^2}+2e^{3\sigma^2}+3e^{2\sigma^2}$
χ^2	$(1-2t)^{-\frac{n}{2}}$	n	$2n$	$\sqrt{\frac{8}{n}}$	$3+\frac{12}{n}$
Exponential	$\frac{1}{1-\sigma t}$	$\sigma+\mu$	σ^2	2	9
Gamma	$(1-\beta t)^{-\alpha}$	$\frac{\alpha}{\beta}$	$\frac{\alpha}{\beta^2}$	$\frac{2}{\sqrt{\alpha}}$	$\frac{6+3\beta}{\beta}$
Inverse-gamma	None	$\frac{\beta}{\alpha-1}$	$\frac{\beta^2}{(\alpha-1)(\alpha-2)}$	$\frac{4\sqrt{\alpha-2}}{(\alpha-1)^2(\alpha-2)}$	$\frac{(3\alpha+15)(\alpha-2)}{(\alpha-3)(\alpha-4)}$
Beta	None	$\frac{\alpha}{\alpha+\beta}$	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$	$\frac{2(\beta-\alpha)\sqrt{\alpha+\beta-1}}{(\alpha+\beta-2)\sqrt{\alpha\beta}}$	$\frac{3(\alpha+\beta+1)(\alpha\beta(\alpha+\beta-6)+2(\alpha+\beta)^2)}{\alpha\beta(\alpha+\beta+2)(\alpha+\beta+3)}$
Rayleigh	$1+e^{\frac{\sigma^2 t^2}{2}}\sigma t\sqrt{\frac{\pi}{2}}\left(erf\left(\frac{\sigma t}{\sqrt{2}}\right)+1\right)$	$\sigma\sqrt{\frac{\pi}{2}}$	$\sigma^2\left(\frac{4-\pi}{2}\right)$	0.631	3.245
Weibull	$\sum_{n=0}^{\infty}\frac{t^n\alpha^n}{n!}\Gamma_n$	$\alpha\Gamma_1$	$\alpha^2\left(\Gamma_2-\Gamma_1^2\right)$	$\frac{\Gamma_3-3\Gamma_2\Gamma_1+2\Gamma_1^3}{(\Gamma_2-\Gamma_1^2)\sqrt{(\Gamma_2-\Gamma_1^2)}}$	$\frac{\Gamma_4-4\Gamma_3\Gamma_1+6\Gamma_2\Gamma_1^2-3\Gamma_1^4}{(\Gamma_2-\Gamma_1^2)^4}$
Gumbel	$e^{\alpha t}\Gamma(1-\beta t)$	$\alpha+\gamma\beta$	$\frac{\beta^2\pi^2}{6}$	$\frac{12\sqrt{6}\zeta(3)}{\pi^3}$	$\frac{12}{5}$

3.18 Summary of the Descriptive Statistics, Moment-Generating Functions, and Moments for the Univariate Distribution

This section is designed to visually summarize the important results from this chapter and to illustrate how the descriptive statistics change between distributions (Table 3.4).

3.19 Summary

In this chapter we have introduced the concepts of discrete and continuous random variables, the probability mass function, and the probability density functions. We have also derived Bayes' theorem from the marginal and conditional PDF or PMF. We have introduced the three descriptive statistics: mode (maximum likelihood state), median (unbiased state), and mean (minimum variance state). We have also introduced the higher-order standardized moments for each of the continuous distribution of variance, skewness, and kurtosis. We have introduced the concept of the moment-generating function for both discrete and continuous distributions.

We have derived the moment-generating functions, if they exist, for all discrete and continuous distributions presented in this chapter. We should note that there are many other distributions involved in geophysical modeling, but if we were to present all of them, then that would be a distribution theory textbook in itself. For the continuous distributions we have presented, we have derived for most of them their first four standardized moments as well as their modes and medians, if they exist in an analytical form.

Given the quite heavy mathematics at the end of the Gumbel distribution, if you made it that far, we now move onto the multivariate distributions that are the foundations of variational, Kalman filtering, particle filters and ensemble based data assimilation schemes.

CHAPTER

Multivariate Distribution Theory

4

Contents

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00008-5

The theory presented in the last chapter was associated with the univariate case, where we only have a single random variable, and it was quite clear that there were complicated expressions and derivations for the three descriptive statistics: median, mode, and mean for several of the distributions. In this chapter we extend the theory to the multivariate case for four of the distributions: Gaussian, lognormal, reverse lognormal, and gamma distribution. We shall introduce the mixed Gaussian-lognormal distribution, along with new distributions that are combinations of a Gaussian, lognormal, and a reverse lognormal distributions.

The multivariate aspect of the distributions in this chapter is referring to a vector, $\mathbf{x}$, whose entries, x_is, $i = 1, 2, \ldots, N$, are random variables from a specific univariate marginal distribution. For the multivariate situation there are relationships that describe the joint behavior between different entries in the vector $\mathbf{x}$.

The basis of variational, Kalman filtering, and ensemble-based data assimilation schemes is a multivariate probability distribution, and as such we need to understand the properties of the descriptive statistics for each distribution, as well as the moments. We start with the definitions for the multivariate distribution and the extension of the descriptive statistics to the multivariate formulation.

4.1 Descriptive Statistics for Multivariate Density Functions

Before we introduce the different definitions for the multivariate descriptive statistics, we need to introduce some new terms that are important in determining different properties of the distributions. The first terms that we introduce are the covariance and joint probability density function. When we deal with bivariate or multivariate distributions, we require the covariance between two random variables. This is given by the following.

Definition 4.1. The covariance, $Cov(X, Y)$, between the two random variables X and Y is given by

$$Cov(X, Y) = \mathbb{E}(XY) - \mathbb{E}(X)\, E(Y), \tag{4.1}$$

where

$$\mathbb{E}(XY) = \int_a^b \int_c^d xy f(x, y)\, dy dx, \tag{4.2}$$

$f(x, y)$ is the **joint probability density function** of $x \in (a, b)$ and $y \in (c, d)$.

During the derivation of Bayes' theorem in the last chapter, we introduced the marginal and conditional PDFs. The definition for the marginal and conditional PDFs extends to the multivariate distributions, where the marginal and the conditional could be univariate or multivariate. To illustrate the link between marginal and conditional distributions for multivariate distributions, we consider the bivariate case; this is where we have two random variables X_1 and X_2, whose associated joint PDF can be factorized as follows:

$$f(x_1, x_2) = f_1(x_1)\, f_2(x_2|x_1) = f_3(x_2)\, f_4(x_1|x_2), \tag{4.3}$$

where $f_{1,3}$ are the marginal distributions for x_1 and x_2, respectively, and $f_{2,4}$ are the conditional distributions for x_2 given x_1 and x_1 given x_2, respectively.

Note: The marginal and conditional PDFs are **not** guaranteed to be of the same distribution type as that of the joint PDF.

4.1.1 Multivariate Moment-Generating Functions

We recall from the univariate theory that the moment-generating function for discrete PMFs and continuous PDFs are given by

$$M_x(t) = \mathbb{E}\left[e^{Xt}\right] \equiv \sum_{\forall x} e^{tx} p_X(x), \tag{4.4a}$$

$$M_x(t) = \mathbb{E}\left[e^{Xt}\right] \equiv \int_a^b e^{xt} f(x)\, dx, \tag{4.4b}$$

respectively.

The extension of the MGF in (4.4a) and (4.4b) is quite straightforward to the bivariate case, where it can be shown that

$$M_{X,Y}(t_1, t_2) = \mathbb{E}\left[e^{t_1 X + t_2 Y}\right] \equiv \sum_{\forall x} \sum_{\forall y} e^{t_1 x + t_2 y} p_{XY}(x, y), \tag{4.5a}$$

$$M_{X,Y}(t_1, t_2) = \mathbb{E}\left[e^{t_1 X + t_2 Y}\right] \equiv \int_a^b \int_c^d e^{t_1 x + t_2 y} f_{XY}(x, y)\, dx dy, \tag{4.5b}$$

where p_{XY} is the joint probability mass function.

To obtain the MGF for the marginal distributions, we evaluate one of the following two expressions:

$$M_X(t_1) = \mathbb{E}\left[e^{t_1 X}\right] \equiv M_{X,Y}(t_1, 0),$$
$$M_Y(t_2) = \mathbb{E}\left[e^{t_2 Y}\right] \equiv M_{X,Y}(0, t_2).$$

If the random variables x and y are independent, then the MGF for the joint PDF becomes

$$M_{X,Y}(t_1, t_2) \equiv M_X(t_1) M_Y(t_2).$$

The expressions for the bivariate MGFs can be extended to the multivariate case and are defined for the discrete and continuous multivariate PMFs and PDFs as

$$M_{\mathbf{X}}(\mathbf{t}) \equiv \mathbb{E}\left[e^{t_1 X_1 + t_2 X_2 + \cdots + t_N X_N}\right] \equiv \mathbb{E}\left[e^{\mathbf{t}^T \mathbf{X}}\right] = \sum_{\forall x_1} \sum_{\forall x_2} \cdots \sum_{\forall x_N} e^{\sum_{i=1}^N t_i x_i} p_{\mathbf{X}}(\mathbf{x}), \tag{4.6a}$$

$$M_{\mathbf{X}}(\mathbf{t}) \equiv \mathbb{E}\left[e^{t_1 X_1 + t_2 X_2 + \cdots + t_N X_N}\right] \equiv \mathbb{E}\left[e^{\mathbf{t}^T \mathbf{X}}\right] = \int_{a_1}^{b_1} \int_{a_2}^{b_2} \cdots \int_{a_N}^{b_N} e^{\sum_{i=1}^N t_i x_i} f_{\mathbf{X}}(\mathbf{x}), \tag{4.6b}$$

respectively, where $\mathbf{t}^T \equiv (t_1, t_2, \ldots, t_N)$, $\mathbf{X} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{pmatrix}$ and $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}$.

4.1.2 Moments of Multivariate Distributions

The non-central moments of the multivariate distributions can be calculated from the multivariate MGFs in the same way for the univariate case, through differentiation, and then setting $t_i = 0$ for $i = 1, 2, \ldots, N$. To obtain the vector of means, we have to differentiate the MGF of the marginal distributions with respect to the t_i that is associated with the specific element of $\mathbf{x}$, x_i, and evaluate the gradient of the MGF at $t_i = 0$. Therefore, for each element of the vector of means we have

$$\begin{aligned}
\mathbb{E}[X_1] &= \frac{\partial}{\partial t_1} M_{\mathbf{X}}(\mathbf{t})\Big|_{t_1=0} (t_1, 0, \ldots, 0), \\
\mathbb{E}[X_2] &= \frac{\partial}{\partial t_2} M_{\mathbf{X}}(\mathbf{t})\Big|_{t_2=0} (0, t_2, \ldots, 0), \\
&\vdots \\
\mathbb{E}[X_N] &= \frac{\partial}{\partial t_N} M_{\mathbf{X}}(\mathbf{t})\Big|_{t_N=0} (0, 0, \ldots, t_N).
\end{aligned}$$

This is an important property, as it implies that we have to find each element of the vector of means separately. If we are lucky enough to have a closed form for the MGF, then it is possible, through vector differentiation, to obtain the vector of means directly. We shall show this for the multivariate Gaussian distribution. If we do not have a closed form for the MGFs, which is true for all forms of the lognormal distribution, then we have to form the marginal distributions and apply the integral form of the expectation operator.

However, the higher order standardized moments are still defined for multivariate distributions and it is possible to obtain a different order of cross moments between a different number of the random variables in $\mathbf{x}$. If we consider the covariance $\mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$ first, then in terms of the MGF for the bivariate case, the product term is obtained through

$$\mathbb{E}[XY] \equiv \frac{\partial^{1+1} M_{X,Y}(t_1, t_2)}{\partial t_1 \partial t_2}\Bigg|_{t_1=t_2=0}. \tag{4.7}$$

This expression can easily be extended to the covariance between different elements, but it can also be used to link different order of moments for different random variables.

If, for example, we wish to evaluate $\mathbb{E}\left[X^3 Y^2\right]$, then this can be achieved through the MGFs as

$$\mathbb{E}\left[X^3 Y^2\right] \equiv \frac{\partial^{3+2} M_{X,Y}(t_1, t_2)}{\partial t_1^3 \partial t_2^2}\Bigg|_{t_1=t_2=0}.$$

This can be generalized to the kth non-central moment for x_1 and the lth non-central moment for x_2 as

$$\mathbb{E}\left[X^k Y^l\right] \equiv \frac{\partial^{k+l} M_{X,Y}(t_1, t_2)}{\partial t_1^k \partial t_2^l}\Bigg|_{t_1=t_2=0}. \tag{4.8}$$

4.1.3 Second-Order Moments: Variance and Covariance

Given the definitions for the moments of a multivariate distribution, we introduce the second-order standardized moments that form a matrix, referred to as the **covariance matrix**, where for the bivariate case the matrix this is

$$\mathbf{\Sigma} \equiv \begin{pmatrix} Var[X_1] & \rho\sqrt{Var[X_1]Var[X_2]} \\ \rho\sqrt{Var[X_1]Var[X_2]} & Var[X_2] \end{pmatrix}, \tag{4.9}$$

where ρ is the correlation coefficient between X_1 and X_2 which is a measure of how much of the variance between the two random variables can be explained by a **LINEAR relationship** and is defined as

$$\rho \equiv \frac{Cov[X_1, X_2]}{\sqrt{Var[X_1]Var[X_2]}}. \tag{4.10}$$

An important property of the covariance is that if the two random variables X_1 and X_2 are independent, then the covariance is equal to zero. A similar relationship exists between the correlation and independence. The independence relationship to correlation is that if two random variables are independent, then they are uncorrelated, so $\rho = 0$. However, if you have no information about the independence of the two variables but you know that their correlation is zero, this **does not** imply that they are independent, just that there does not exist a linear relationship between the two random variables.

4.1.4 Third-Order Moments: Skewness and Co-Skewness

While there are many different combinations of the higher-order moments for the multivariate PMFs and PDFs, we only consider the definition of multivariate skewness, and in the next subsection multivariate kurtosis, which is an indicator that the data we are using is not multivariate Gaussian distributed if the equivalent of the univariate skewness condition of $\beta_1 = 0$ is not true for all of the marginal distributions.

From univariate theory, we know that the definition of skewness is

$$\beta_1 = \frac{\mathbb{E}\left[(X - \mathbb{E}[X])^3\right]}{\left(\mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2\right)^{\frac{3}{2}}}, \tag{4.11}$$

and we also know that the non-central and standardized moments are calculated from each of the marginal distributions. Therefore, for a bivariate distribution we would have two coefficients of skewness associated with each marginal, denoted here as $\beta_{1,1}$ and $\beta_{1,2}$. However, as we saw from the second-order moments, we can have a cross term, called the covariance for the second moment, and as we suggested in the MGF section, it is possible to have higher-order co-moments. This is true for skewness and is referred to as **co-skewness**, which we shall denote as Cos.

Co-skewness came about through the work of economists in the mid-1970s as a way to examine risk in stock market investments. In probability and statistics it is a measure of how much two or three random variables change with each other, and is referred to as the third order cross-central-moment.

There are two definitions of co-skewness for the bivariate case, which are given by

$$Cos\left[X_1^2, X_2\right] \equiv \beta_{1,x_1^2,x_2} = \frac{\mathbb{E}\left[(X_1 - \mathbb{E}[X_1])^2 (X_2 - \mathbb{E}[X_2])\right]}{\sigma_{X_1}^2 \sigma_{X_2}}, \tag{4.12a}$$

$$Cos\left[X_1, X_2^2\right] \equiv \beta_{1,x_1,x_2^2} = \frac{\mathbb{E}\left[(X_1 - \mathbb{E}[X_1]) (X_2 - \mathbb{E}[X_2])^2\right]}{\sigma_{X_1} \sigma_{X_2}^2}. \tag{4.12b}$$

If we expand the expectation operator in (4.12a), we obtain

$$\begin{aligned} &Cos\left[X_1, X_2\right] \\ &= \frac{\mathbb{E}\left[\left(X_1^2 X_2 - 2X_1 X_2 \mathbb{E}[X_1] + X_2 (\mathbb{E}[X_1])^2 - X_1^2 \mathbb{E}[X_2] + 2X_1 \mathbb{E}[X_1]\mathbb{E}[X_2] - \mathbb{E}[X_2](\mathbb{E}[X_1])^2\right)\right]}{\sigma_{X_1^2} \sigma_{X_2}}, \\ &= \frac{\mathbb{E}\left[X_1^2 X_2\right] - 2\mathbb{E}[X_1 X_2]\mathbb{E}[X_1] + \mathbb{E}[X_2](\mathbb{E}[X_1])^2 - \mathbb{E}\left[X_1^2\right]\mathbb{E}[X_2] + 2(\mathbb{E}[X_1])^2 \mathbb{E}[X_2] - (\mathbb{E}[X_1])^2 \mathbb{E}[X_2]}{\sigma_{X_1}^2 \sigma_{X_2}}. \end{aligned} \tag{4.13}$$

While the expression in (4.13) may look daunting, upon inspection we can see some familiar terms there. We just need to factorize and notice that there are two terms that cancel each other. Thus it is possible to simplify (4.13) to

$$Cos\left[X_1, X_2\right] \equiv \frac{\mathbb{E}\left[X_1^2 X_2\right] - \mathbb{E}\left[X_1^2\right]\mathbb{E}[X_2] - 2Cov[X_1, X_2]\mathbb{E}[X_1]}{\sigma_{X_1}^2 \sigma_{X_2}}. \tag{4.14}$$

Finally, recognizing that the first two terms in (4.14) are equivalent to the definition of covariance, but here it is between X_1^2 with X_2, we can obtain a final expression for co-skewness in terms of covariances and a mean as

$$Cos\left[X_1^2, X_2\right] = \frac{Cov\left[X_1^2, X_2\right] - 2Cov[X_1, X_2]\mathbb{E}[X_1]}{\sigma_{X_1^2} \sigma_{X_2}}, \tag{4.15}$$

which now appears less daunting than (4.13). An important property to identify here is that if X_1 and X_2 are independent, then they will have a zero co-skewness coefficient.

The derivation above can be applied to (4.12b), which results in

$$Cos\left[X_1, X_2^2\right] = \frac{Cov\left[X_1, X_2^2\right] - 2Cov[X_1, X_2]\mathbb{E}[X_2]}{\sigma_{X_1} \sigma_{X_2^2}}. \tag{4.16}$$

Note: (4.15) and (4.16) are not the same.

As with the second moment, we can form a matrix to store the third-order co-moments, which we shall refer to as the co-skewness matrix. This is defined as

$$\mathbb{M}_3 = \begin{pmatrix} \beta_{x_1^3} & \beta_{x_1^2,x_2} \\ \beta_{x_1,x_2^2} & \beta_{x_2^3} \end{pmatrix}. \tag{4.17}$$

Therefore, the diagonal entries of the co-skewness matrix for the bivariate distributions are the coefficients of skewness of the marginal distributions. Given the property of (4.15) and (4.16), which are the off-diagonal entries in (4.17), we must note that this matrix is **not** symmetric.

However, when considering a multivariate distribution higher than order two, the co-skewness matrix has a different form. If we consider a generic trivariate distribution, then the definition of the co-skewness matrix is

$$\mathbb{M}_3 \equiv \mathbb{E}\left[(\mathbf{X}-\mathbb{E}[\mathbf{X}])(\mathbf{X}-\mathbb{E}[\mathbf{X}])^T \otimes (\mathbf{X}-\mathbb{E}[\mathbf{X}])\right] \equiv \left\{\beta_{x_i,x_j,x_k}\right\}, \tag{4.18}$$

$$\left\{\beta_{x_i,x_j,x_k}\right\} \equiv \mathbb{E}\left[(x_i-\mathbb{E}[x_i])\left(x_j-\mathbb{E}\left[x_j\right]\right)(x_k-\mathbb{E}[x_k])\right], \tag{4.19}$$

where $\otimes$ is the Kronecker product operator, which is an element multiplying a matrix; for example,

$$\mathbf{A}=\begin{pmatrix}\alpha & \beta\\ \gamma & \delta\end{pmatrix}, \quad \mathbf{B}=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}, \quad \mathbf{A}\otimes\mathbf{B}=\begin{pmatrix}\alpha & 2\alpha & \beta & 2\beta\\ 3\alpha & 4\alpha & 3\beta & 4\beta\\ \gamma & 2\gamma & \delta & 2\delta\\ 3\gamma & 4\gamma & 3\delta & 4\delta\end{pmatrix}.$$

Therefore, we can define the co-skewness matrix as a set of three 3×3 arrays as

$$\begin{aligned}\mathbb{M}_3 &\equiv \left(\begin{array}{ccc|ccc|ccc}\beta_{111} & \beta_{112} & \beta_{113} & \beta_{211} & \beta_{212} & \beta_{213} & \beta_{311} & \beta_{312} & \beta_{313}\\ \beta_{121} & \beta_{122} & \beta_{123} & \beta_{221} & \beta_{222} & \beta_{223} & \beta_{321} & \beta_{322} & \beta_{323}\\ \beta_{131} & \beta_{132} & \beta_{133} & \beta_{231} & \beta_{232} & \beta_{233} & \beta_{331} & \beta_{332} & \beta_{333}\end{array}\right)\\ &= \left(\boldsymbol{\beta}_{1jk}\boldsymbol{\beta}_{2jk}\boldsymbol{\beta}_{3jk}\right),\end{aligned} \tag{4.20}$$

where $\boldsymbol{\beta}_{1jk}$ denotes the matrix whose entries are $\left\{\beta_{1jk}\right\}$ for $j,k=1,2,3$.

We should note here that there is a third expression for co-skewness, which is a function of three random variables as

$$\beta_{123} \equiv \frac{\mathbb{E}[(X_1-E[X_1])(X_2-E[X_2])(X_3-E[X_3])]}{\sigma_1\sigma_2\sigma_3}. \tag{4.21}$$

Upon multiplying out the brackets and applying the expectation operator in (4.21), we obtain

$$\beta_{123} = \frac{\mathbb{E}[X_1X_2X_3]-\mathbb{E}[X_1]\mathbb{E}[X_2]\mathbb{E}[X_3]-Cov[X_1,X_2]\mathbb{E}[X_3]-Cov[X_1,X_3]\mathbb{E}[X_2]-Cov[X_2,X_3]\mathbb{E}[X_1]}{\sigma_1\sigma_2\sigma_3}. \tag{4.22}$$

Two important features to note about the expression in (4.22): first, if say $X_3=X_1$, then at first glance it would appear that the expression in (4.22) does not collapse to (4.15). However, if we evaluate (4.22) with $X_3=X_1$, then we obtain

$$\beta_{121} = \frac{\mathbb{E}\left[X_1^2X_2\right]-(\mathbb{E}[X_1])^2\mathbb{E}[X_2]-2Cov[X_1,X_2]\mathbb{E}[X_1]-Var[X_1]\mathbb{E}[X_2]}{\sigma_1^2\sigma_2}, \tag{4.23}$$

where $Cov[X_1,X_1]\equiv Var[X_1]$. We do not quite have the correct terms for the covariance of X_1^2 with X_2, and we certainly do not have any terms with respect to variance in (4.15). The trick is to rearrange the definition of variance to isolate $(\mathbb{E}[X_1])^2$ as $\mathbb{E}\left[X_1^2\right]-Var[X_1]$ and to substitute this expression for $(\mathbb{E}[X_1])^2$, which cancels the variance term and gives us the correct term to form $Cov\left[X_1^2,X_2\right]$.

The second important property to note is that if any of the three random variables are independent of the other two, then the coskewness for this triplet is equal to zero, even if the remaining two random variables are not independent of each other. To see this property, let us say that X_3 is independent of X_1 and X_2, but X_1 and X_2 are not independent of each other. The last two covariance terms in (4.22) are then equal to zero, but we are still left with the covariance term of X_1 with X_2. However, if we consider the first term in (4.22), then because X_3 is independent of both X_1 and X_2 the first term in (4.22) becomes $\mathbb{E}[X_1X_2]\mathbb{E}[X_3]$. Factorizing $\mathbb{E}[X_3]$ from the first two remaining terms leaves the expression for the covariance of X_1 and X_2, but this is of the opposite sign of the remaining covariance term, and hence they cancel each other, resulting in $\beta_{123} = 0$.

4.1.5 Fourth-Order Moments: Kurtosis and Co-kurtosis

From univariate theory, we know that the definition of kurtosis is

$$\beta_2 = \frac{\mathbb{E}\left[(X - \mathbb{E}[X])^4\right]}{\left(\mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2\right)^2}. \tag{4.24}$$

As with skewness, kurtosis is calculated for each of the marginal distributions of the multivariate distribution being considered. However, for co-kurtosis there are three possible permutations for a bivariate distribution that can be classified as co-kurtosis:

$$Cok[X_1, X_1, X_1, X_2] = \frac{\mathbb{E}\left[(X_1 - \mathbb{E}[X_1])^3 (X_2 - \mathbb{E}[X_2])\right]}{\sigma_{X_1}^3 \sigma_{X_2}},$$

$$Cok[X_1, X_1, X_2, X_2] = \frac{\mathbb{E}\left[(X_1 - \mathbb{E}[X_1])^2 (X_2 - \mathbb{E}[X_2])^2\right]}{\sigma_{X_1}^2 \sigma_{X_2}^2},$$

$$Cok[X_1, X_2, X_2, X_2] = \frac{\mathbb{E}\left[(X_1 - \mathbb{E}[X_1]) (X_2 - \mathbb{E}[X_2])^3\right]}{\sigma_{X_1} \sigma_{X_2}^3}.$$

These are collectively referred to as **co-kurtoses**.

As with co-skewness, if we were to expand the expectation operators above, then we could express the co-kurtoses as functions of variances of different powers of the random variables as

$$Cok[X_1, X_1, X_1, X_2] \equiv \frac{Cov\left[X_1^3, X_2\right] - 3Cov\left[X_1^2, X_2\right]\mathbb{E}[X_1] + 3Cov[X_1, X_2]\mathbb{E}\left[X_1^2\right]}{\sigma_1^3 \sigma_2}, \tag{4.25a}$$

$$Cok[X_1, X_2, X_2, X_2] \equiv \frac{Cov\left[X_1, X_2^3\right] - 3Cov\left[X_1, X_2^2\right]\mathbb{E}[X_2] + 3Cov[X_1, X_2]\mathbb{E}\left[X_2^2\right]}{\sigma_1 \sigma_2^3}. \tag{4.25b}$$

If we were to go to a trivariate or a quadvariate distribution, then we could have co-kurtoses that are functions of covariance, where in two of the expressions above we have the skewness of the marginal distribution interacting with the standard deviation of the other random variables.

It is possible to define a co-kurtosis matrix, but the expression is quite cumbersome and at the moment is not used much in data assimilation. However, there has been work recently on a multivariate Gaussian formulation to try to adapt the systems to fit to these higher-order co-moments [174,175].

4.1.6 Mode of Multivariate Distribution

The definition of the multivariate modes is the simplest of all of the three descriptive statistics to evaluate or approximate. We saw for the univariate case that if a distribution is unimodal, then the associated value of the random variable x such that the probability density function was at it maximum could be found by simply differentiating the density function and finding the zero of the gradient.

The theory extends to the multivariate case, as a result of calculus theory extending to vectors. Thus, we can state that the definition of the multivariate mode/maximum likelihood state as:

Definition 4.2. Given a vector $\mathbf{x}$ whose components are random variables such that the vector follows a multivariate continuous probability density function $f(\mathbf{x})$, then the mode, $\mathbf{x}_{mode}$, of the multivariate distribution is such that

$$\mathbf{x}_{mode} \text{ such that } \left.\frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{mode}} = 0. \tag{4.26}$$

The other important property of the multivariate mode for multivariate unimodal distributions is that it is unique. This means that there is only one set of values for the x_is in $\mathbf{x}$ such that (4.26) is true. The reason why this property is highlighted will become clear after we have introduced the definition of the multivariate median.

4.1.7 Median of Multivariate Distribution

As we saw for the univariate case, the median is the unbiased estimator, and this property carries over to the multivariate situation. However, being unbiased in multivariate distribution theory has its problems. We have seen that the unbiased statistic is the value for the random variable such that the cumulative density function evaluated at that value is equal to 0.5. This implies that the values of the distribution lie equally on either side.

When considering the multivariate situation, the definition for the univariate median extends to each component of the vector of random variables. Thus, we can define the median of a multivariate distribution as:

Definition 4.3. Given a vector $\mathbf{x}$ whose components are random variables such that the vector follows a multivariate continuous probability density function $f(\mathbf{x})$, then the median $\mathbf{x}_{med}$ of the multivariate distribution is such that

$$\mathbf{x}_{med} \text{ such that } \int_{x_{1l}}^{x_{m1}} \int_{x_{2l}}^{x_{m2}} \cdots \int_{x_{Nl}}^{x_{mN}} f(\mathbf{x})\, dx_1 dx_2 \ldots dx_N = \frac{1}{2}, \tag{4.27}$$

where the x_{il} refer to the lower limit of the range of the distribution for that component for $i = 1, 2, \ldots, N$ and x_{mi} are the component of the median vector.

At first, the unbiased estimator, median, looks like a desirable statistic to seek; however, as we look more closely at the definition in (4.27), we see that this statistic is **non-unique**. The non-uniqueness arises from the fact that given any values for any one component of the integral, we can always find

the values for the remaining components such that the integral condition in (4.27) is satisfied. If we consider the simple two-dimensional case where $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, then if we set x_1 to a specific values then there exists a value for x_2 such that (4.27) is satisfied. The same is true if we have a specific value for x_2: a value will exist for x_1 such that (4.27) is satisfied.

Therefore, for each value of x_1 there exists a value for x_2 such that (4.27) is satisfied, and hence the non-uniqueness. The same argument applies in higher dimensions.

We now move on to consider some of the statistics for a selection of multivariate PDFs.

4.2 Gaussian Distribution

The most commonly used distribution to approximate many different errors associated with different geophysical modeling and of the errors associated with observing different geophysical flows is the multivariate Gaussian distribution. In this section we shall first introduce the bivariate Gaussian distribution and present its marginal and conditional PDFs as well as deriving its means, variance, and covariance. We shall also derive the median and the mode for the bivariate Gaussian distribution, but for the latter descriptive statistic this derivation will be in the multivariate formulation, as it is obtained through vector differentiation, which is the same if the distribution has two or more random variables. Following the presentation of the bivariate Gaussian distribution, we shall present the multivariate Gaussian distribution and its factorization into different orders of marginal and conditional PDFs.

4.2.1 Bivariate Gaussian Distribution

The random variables X_1 and X_2 that are jointly distributed with a **bivariate Gaussian**, $BG(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu}^T = (\mu_1, \mu_2)$ is the vector of the means of the random variables X_1 and X_2, respectively, and

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

is the covariance matrix for the bivariate Gaussian where σ_1 and σ_2 are the standard deviations of X_1 and X_2, respectively, and ρ is the correlation coefficient as defined earlier, follow a bivariate probability density function given by

$$\boldsymbol{x} \sim BG(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \Rightarrow f(x_1, x_2) = \frac{1}{2\pi |\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})}, \tag{4.28}$$

$$f(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{(1-\rho^2)}} e^{-\frac{1}{2}\left(\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2 - 2\rho\left(\frac{x_1-\mu_1}{\sigma_1}\right)\left(\frac{x_2-\mu_2}{\sigma_2}\right) + \left(\frac{x_2-\mu_2}{\sigma_2}\right)^2\right)}, \tag{4.29}$$

where $x_1 \in (-\infty, \infty)$ and $x_2 \in (-\infty, \infty)$.

Due to the symmetry of the bivariate Gaussian distribution, we can factorize this distribution into the product of the marginal distribution of x_1 and the conditional distribution of x_2 given x_1, or vice versa. In the derivation of the first set of moments, we use the factorization

$$\begin{aligned} BG(x_1, x_2) &= \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1^2}\right)^2} \frac{1}{\sqrt{2\pi}\sigma_2\sqrt{(1-\rho^2)}} e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_2^2(1-\rho^2)}}, \\ &= f_1(x_1)\, f(x_2 \,|\, x_1), \end{aligned} \tag{4.30}$$

where $\mu_{x_2|x_1}$ is the **conditional mean** and $\sigma_2^2(1-\rho^2)$ is the **conditional variance**. The conditional mean is given by

$$\mu_{x_2|x_1} = \mu_2 + \rho\frac{\sigma_2}{\sigma_1}(x_1 - \mu_1). \tag{4.31}$$

The conditional distribution defines a univariate Gaussian distribution $G\left(\mu_{x_2|x_1}, \sigma_2^2(1-\rho^2)\right)$. Given the formulation just described we now derive the expectation of x_1:

$$\begin{aligned} \mathbb{E}(X_1) &= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_2\sqrt{(1-\rho^2)}}\int_{-\infty}^{\infty} e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_2^2(1-\rho^2)}} dx_2\right) dx_1, \\ &= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \left(\int_{-\infty}^{\infty} G\left(\mu_{x_2|x_1}, \sigma_2^2\left(1-\rho^2\right)\right) dx_2\right) dx_1, \\ &= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} dx_1, \\ &= \mu_1. \end{aligned}$$

Therefore, the expectations of the marginal distributions are μ_1 or μ_2 due to the symmetry of the argument above. For the derivation of the variances, we follow the same argument as set out above for $\mathbb{E}\left(X_1^2\right)$ by using the fact that the second integral in the derivation is the cumulative density function of a Gaussian distribution.

Exercise 4.4. *Derive the second uncentered moment for x_1 given the factorization of the bivariate Gaussian distribution in* (4.30).

Below we show the derivation of the expectation of x_2, but instead of swapping x_2 for x_1 in the marginal distribution, we shall obtain the expressing by integrating the conditional distribution:

$$\mathbb{E}(X_2) = \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_2\sqrt{(1-\rho^2)}}\int_{-\infty}^{\infty} x_2 e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_2^2(1-\rho^2)}} dx_2\right) dx_1,$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \mu_{x_2|x_1} dx_1,\\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} \left(\mu_2 + \rho\frac{\sigma_2}{\sigma_1}(x_1-\mu_1)\right) e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} dx_1,\\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} (\mu_2 + \rho\sigma_2 z_1)\, e^{-\frac{z_1^2}{2}} dz_1,\\
&= \mu_2,
\end{aligned}
$$

where $z_1 \equiv \dfrac{x_1-\mu_1}{\sigma_1}$.

For the square of the second order non-central moment we have

$$
\begin{aligned}
\mathbb{E}\left(X_2^2\right) &= \frac{1}{\sqrt{\pi}\sigma_1}\int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \left(\frac{1}{\sigma_2\sqrt{2\pi}\sqrt{(1-\rho^2)}}\int_{-\infty}^{\infty} x_2^2 e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_2^2(1-\rho^2)}} dx_2\right) dx_1,\\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2}\\
&\quad\times\left(\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\left(\sigma_2^2\left(1-\rho^2\right)z_2^2 + 2\sigma_2\sqrt{(1-\rho^2)}\mu_{x_2|x_1}w_2 + \mu_{x_2|x_1}^2\right)e^{-\frac{z_2^2}{2}} dz_2\right) dx_1,\\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} e^{-\frac{1}{2}\left(d\frac{x_1-\mu_1}{\sigma_1}\right)^2}\left(\sigma_2^2\left(1-\rho^2\right)+\mu_{x_2|x_1}^2\right)dx_1,\\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty}\left(\sigma_2^2\left(1-\rho^2\right)+\mu_2^2+2\mu_2\rho\frac{\sigma_2}{\sigma_1}(x_1-\mu_1)+\rho^2\frac{\sigma_2^2}{\sigma_1^2}(x_1-\mu_1)^2\right)e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} dx_1,\\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\left(\mu_2^2+\sigma_2^2\left(1-\rho^2\right)+2\rho\sigma_2 w_1+\rho^2\sigma_2^2 z_2^2\right)e^{-\frac{z_1^2}{2}} dz_1,\\
&= \mu_2^2+\sigma_2^2\left(1-\rho^2\right)+\rho^2\sigma_2^2 = \mu_2^2+\sigma_2^2.
\end{aligned}
$$

The final moment that we need is the covariance, given by

$$
\mathbb{E}(X_1X_2) = \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2}\left(\frac{x_2}{\sqrt{2\pi}\sqrt{(1-\rho^2)}\sigma_2}\int_{-\infty}^{\infty} e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_2^2(1-\rho^2)}} dx_2\right) dx_1,
$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \mu_{x_2|x_1} dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_{-\infty}^{\infty} \left(\mu_2 + \rho\frac{\sigma_2}{\sigma_1}(x_1-\mu_1)\right) x_1 e^{-\frac{1}{2}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} dx_1, \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} (\mu_2 + \rho\sigma_2 z_1)(\mu_1 + \sigma_1 z_1) e^{-\frac{z_1^2}{2}} dz_1, \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \left(\mu_1\mu_2 + \mu_2\sigma_1 z_1 + \rho\mu_1\sigma_2 z_1 + \rho\sigma_1\sigma_2 z_1^2\right) e^{-\frac{z_1^2}{2}} dz_1, \\
&= \mu_1\mu_2 + \rho\sigma_1\sigma_2.
\end{aligned} \tag{4.32}
$$

Recalling that the covariance is $\mathbb{E}(X_1X_2) - \mathbb{E}(X_1)\mathbb{E}(X_2)$, this then gives $Cov(X_1X_2) = \mu_1\mu_2 + \rho\sigma_1\sigma_2 - \mu_1\mu_2 = \rho\sigma_1\sigma_2$.

4.2.2 Medians of the Bivariate Gaussian Distribution

As mentioned earlier in this chapter, there are more than one set of values for the random variables that meet the definition for a median of a multivariate distribution. For the bivariate Gaussian case, we have to evaluate the integrals

$$
\begin{aligned}
\mathbf{x}_{median}: \quad & \int_{-\infty}^{x_{1,med}}\int_{-\infty}^{x_{2,med}} \frac{1}{\sqrt{2\pi}\sigma_1\sigma_2\sqrt{(1-\rho^2)}} e^{-\frac{1}{2(1-\rho^2)}\left(\frac{x_1-\mu_1}{\sigma_1}\right)^2} \\
& -2\rho\left(\frac{x_1-\mu_1}{\sigma_1}\right)\left(\frac{x_2-\mu_2}{\sigma_2}\right) + \left(\frac{x_2-\mu_2}{\sigma_2}\right)^2 dx_1 dx_2 = \frac{1}{2}.
\end{aligned} \tag{4.33}
$$

By factorizing (4.33) into its marginal and condition univariate Gaussian distributions, we can see by component that the median is also equal to the vector of means μ.

4.2.3 Mode of the Bivariate Lognormal

As mentioned earlier, when introducing the definitions of the multivariate descriptive statistics, the derivation of the maximum likelihood state, or the bivariate mode, is a straightforward extension from the univariate to the bivariate case. It can be achieved by considering the matrix definition of the bivariate Gaussian distribution and then differentiating with respect to the vector of random variables. Therefore we have the definition of the bivariate form of the Gaussian distribution as

$$
BG(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{2\pi}\,|\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}}, \tag{4.34}
$$

where we now differentiate (4.34) with respect to $\mathbf{x}$ and set the Jacobian to zero, which is

$$\left.\frac{dBG(\boldsymbol{\mu},\boldsymbol{\Sigma})}{d\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{mode}} = -\frac{1}{\sqrt{2\pi}\,|\boldsymbol{\Sigma}|^{\frac{1}{2}}}\left(\boldsymbol{\Sigma}^{-1}(\mathbf{x}_{mode}-\boldsymbol{\mu})\right)e^{\left\{-\frac{1}{2}(\mathbf{x}_{mode}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}_{mode}-\boldsymbol{\mu})\right\}} = \mathbf{0}. \quad (4.35)$$

For the expression in (4.35) to be equal to zero, we have to consider each of the terms that are functions of $\mathbf{x}$. If we consider the exponential term, we see that it can never be equal to zero except as $\mathbf{x}$ tends to $\pm\infty$. Therefore we are left with

$$-\boldsymbol{\Sigma}^{-1}(\mathbf{x}_{mode}-\boldsymbol{\mu}) = \mathbf{0} \Rightarrow \mathbf{x}_{mode} = \boldsymbol{\mu}. \quad (4.36)$$

Thus the mode of the bivariate Gaussian distribution is equal to the vector of means, $\boldsymbol{\mu}$.

We now move on to the definition of the multivariate Gaussian distribution, which is the foundation of the variational, Kalman filter and ensemble-based data assimilation systems.

4.2.4 Multivariate Gaussian Distribution

Given a vector, $\mathbf{x}$, of N random variables, where $x_i \in (-\infty,\infty)$ for $i = 1, 2, \ldots, N$, the associated multivariate Gaussian distribution is defined as

$$MG(\boldsymbol{\mu},\boldsymbol{\Sigma}) \equiv \frac{1}{(2\pi)^{\frac{N}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}}e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}, \quad (4.37)$$

where the covariance matrix for the multivariate case is defined by

$$\boldsymbol{\Sigma} \equiv \begin{pmatrix} Var[X_1] & Cov[X_1,X_2] & Cov[X_1,X_3] & \cdots & Cov[X_1,X_N] \\ Cov[X_2,X_1] & Var[X_2] & Cov[X_2,X_3] & \cdots & Cov[X_2,X_N] \\ Cov[X_3,X_1] & Cov[X_3,X_2] & Var[X_3] & \ddots & Cov[X_3,X_N] \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ Cov[X_N,X_1] & Cov[X_N,X_2] & Cov[X_N,X_3] & \cdots & Var[X_N] \end{pmatrix}, \quad (4.38)$$

$$\equiv \begin{pmatrix} \sigma_1^2 & \rho_{1,2}\sigma_1\sigma_2 & \rho_{1,3}\sigma_1\sigma_3 & \cdots & \rho_{1,N}\sigma_1\sigma_N \\ \rho_{2,1}\sigma_2\sigma_1 & \sigma_2^2 & \rho_{2,3}\sigma_2\sigma_3 & \cdots & \rho_{2,N}\sigma_2\sigma_N \\ \rho_{3,1}\sigma_3\sigma_1 & \rho_{3,2}\sigma_3\sigma_2 & \sigma_3^2 & \ddots & \rho_{3,N}\sigma_3\sigma_N \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{N,1}\sigma_N\sigma_1 & \rho_{N,2}\sigma_N\sigma_2 & \rho_{N,3}\sigma_N\sigma_3 & \cdots & \sigma_N^2 \end{pmatrix}, \quad (4.39)$$

where $\rho_{i,j}$ is the correlation coefficient between random variables x_i and x_j.

As we saw with the bivariate Gaussian distribution, it is possible to factorize the multivariate Gaussian distribution into marginal and conditional distributions of varying dimensions. We start by

expressing the vector of means and the covariance matrix in terms of sub-vectors and sub-matrices as

$$\begin{aligned} \boldsymbol{\mu} &\equiv \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{pmatrix}, & \boldsymbol{\Sigma} &\equiv \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix} \\ \boldsymbol{\mu} &\equiv \begin{bmatrix} p \times 1 \\ (N-p) \times 1 \end{bmatrix}, & \boldsymbol{\Sigma} &\equiv \begin{bmatrix} p \times p & p \times (N-p) \\ (N-p) \times p & (N-p) \times (N-p) \end{bmatrix}. \end{aligned} \tag{4.40}$$

Given the partitions in (4.40), then it is possible to define the conditional mean, and covariance matrix as

$$\overline{\boldsymbol{\mu}} \equiv \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1), \quad \overline{\boldsymbol{\Sigma}} \equiv \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}. \tag{4.41}$$

Therefore, the marginal distribution is a multivariate Gaussian distribution, $MG(\boldsymbol{\mu}_{11}, \boldsymbol{\Sigma}_{11})$, and the multivariate conditional distribution is also a multivariate Gaussian distribution such that $MG\left(\overline{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}}\right)$.

Therefore, given the factorization of the multivariate Gaussian distribution into a multivariate marginal times a multivariate conditional distribution, we can find the first moments of each of the random variables through factorizing the distribution into the product of a univariate Gaussian marginal and a $N \times 1$ conditional multivariate Gaussian distribution.

We have already shown that derivation for the multivariate mode for the Gaussian distribution due to considering the vectorial form of the bivariate Gaussian distribution. The derivation for the median for the bivariate Gaussian is also easily extended to the multivariate situation.

We now move on to our next probability distribution: the multivariate lognormal distribution.

4.3 Lognormal Distribution

The multivariate lognormal distribution is used to describe a multivariate relationship between positive definite random variables. Definitions of the multivariate forms of the lognormal distribution are quite similar in appearance to those of the multivariate Gaussian, but now involve the logarithm of the random variables, and a scaling term, x_i^{-1}, in front of the exponential. As with the Gaussian distribution, we shall start with the bivariate form of the lognormal distribution and then move on to the multivariate version.

4.3.1 Bivariate Lognormal Distribution

Given two positive definite random variables x_1 and x_2 such that $x_1, x_2 \in (0, \infty)$, then the bivariate lognormal distribution that describes the relationship between these two random variables is defined as

$$\begin{aligned} BLN(\boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L) \equiv & \frac{1}{\sqrt{2\pi}\sigma_{1_L}\sigma_{2_L}\sqrt{\left(1-\rho_L^2\right)}} \frac{1}{x_1 x_2} \\ & e^{-\frac{1}{2\left(1-\rho_L^2\right)}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}}\right)^2 - 2\rho_L\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}}\right)\left(\frac{\ln x_2 - \mu_{2_L}}{\sigma_{2_L}}\right) + \left(\frac{\ln x_2 - \mu_{2_L}}{\sigma_{2_L}}\right)^2}, \end{aligned} \tag{4.42}$$

where $\mu_{i_L} \equiv \mathbb{E}[\ln X_i]$, $\sigma^2_{i_L} \equiv \mathbb{E}\left[(\ln X_i)^2\right] - (\mathbb{E}[\ln X_i])^2$, and the correlation coefficient is defined as

$$\rho_L \equiv \frac{Cov[\ln X_1, \ln X_2]}{\sqrt{Var[\ln X_1]\, Var[\ln X_2]}}.$$

We should recall from the univariate lognormal distribution that all of the statistics involved in the definition of the multivariate distribution are in terms of $\ln x_i$ and **not** x_i.

We now consider the moments of the bivariate lognormal distribution.

4.3.2 Moments of the Bivariate Lognormal Distribution

As with the bivariate Gaussian, it is also possible to factorize the bivariate lognormal distribution into the product of univariate marginal distribution and a univariate conditional distribution as

$$BLN(x_1, x_2) = \frac{1}{\sqrt{2\pi}\sigma_{1_L} x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1,L}}\right)^2} \frac{1}{\sqrt{2\pi}\sigma_{2_L}\sqrt{(1-\rho_L^2)}x_2} e^{-\frac{(\ln x_2 - \mu_{x_2|x_1})^2}{2\sigma^2_{2_L}(1-\rho_L^2)}}, \tag{4.43}$$

where both the marginal and conditional distribution in (4.43) are univariate lognormal distributions. The conditional distribution is given by $LN\left(\mu_{x_2|x_1}, \sigma^2_{2_L}(1-\rho_L^2)\right)$, where the lognormal conditional mean $\mu_{x_2|x_1}$ is defined as

$$\mu_{x_2|x_1} = \mu_{2_L} + \rho_L \frac{\sigma_{2_L}}{\sigma_{1_L}}(\ln x_1 - \mu_{1_L}). \tag{4.44}$$

Given the factorizations above, it is possible to show that the expectations of X_1 is

$$\begin{aligned}
\mathbb{E}(X_1) &= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma^2_{1_L}}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_{2_L}\sqrt{(1-\rho_L^2)}} \int_0^\infty \frac{1}{x_2} e^{-\frac{(\ln x_2 - \mu_2)^2}{2\sigma^2_{2_L}(1-\rho_L^2)}} dx_2 \right) dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma^2_{1_L}}\right)^2} \int_0^\infty LN\left(\mu_{x_2|x_1}, \sigma^2_{2_L}(1-\rho_L^2)\right) dx_2 dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma^2_{1_L}}\right)^2} dx_1, \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{\sigma_{1_L} w_1 + \mu_{1_L}} e^{-\frac{w_1^2}{2}} dw_1, \\
&= e^{\mu_{1_L} + \frac{\sigma^2_{1_L}}{2}}.
\end{aligned} \tag{4.45}$$

To obtain the variance of X_1, we require $\mathbb{E}\left[X_1^2\right]$, which is

$$\mathbb{E}\left(X_1^2\right) = \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty x_1 e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_2\sqrt{\left(1-\rho_L^2\right)}} \int_0^\infty \frac{1}{x_2} e^{-\frac{\left(\ln x_2 - \mu_{2_L}\right)^2}{2\sigma_{2_L}^2\left(1-\rho_L^2\right)}} dx_2 \right) dx_1,$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty x_1 e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} dx_1$$

$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{2\sigma_{1_L} w_1 + 2\mu_{1_L}} e^{-\frac{w_1^2}{2}} dw_1,$$

$$= e^{2\sigma_{1_L}^2 + 2\mu_{1_L}}.$$

To find the expressions for $\mathbb{E}[X_2]$ and $\mathbb{E}\left[X_2^2\right]$, we could rearrange the derivation above, where we would replace the subscript 1 with the subscript 2. However, here we shall show the derivation by evaluating the expectations of the conditional PDF. The reason for showing the derivation this way is, as seen for the bivariate Gaussian distribution, that we need to be able to integrate the conditional PDF for the expectation of the product of X_1 and X_2:

$$\mathbb{E}(X_2) = \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_{2_L}\sqrt{\left(1-\rho_L^2\right)}} \int_0^\infty e^{-\frac{\left(\ln x_2 - \mu_{x_2|x_1}\right)^2}{2\sigma_{2_L}^2\left(1-\rho_L^2\right)}} dx_2 \right) dx_1,$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{\sigma_{2_L}\sqrt{\left(1-\rho_L^2\right)} w_2 + \mu_{x_2|x_1}} e^{-\frac{w_1^2}{2}} dw_1 dx_1,$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} e^{\mu_{x_2|x_1} + \frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2}} dx_1,$$

$$= e^{\frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2} + \mu_{2_L}} \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{\rho_L \frac{\sigma_{2_L}}{\sigma_{1_L}}\left(\ln x_1 - \mu_{1_L}\right)} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} dx_1,$$

$$= e^{\frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2} + \mu_{2_L}} \frac{1}{\sqrt{2\pi}\sigma_1} \int_{-\infty}^\infty e^{\rho_L \sigma_{2_L} w_1} e^{-\frac{w_1^2}{2}} dw_1,$$

$$= e^{\frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2} + \mu_{2_L} + \frac{\rho_L^2\sigma_{2_L}^2}{2}},$$

$$= e^{\mu_{2_L} + \frac{\sigma_{2_L}^2}{2}}. \tag{4.46}$$

Therefore the marginal expectation of X_2 is that of a lognormally distributed random variable. Moving on to the square of the second order non-central moment of X_2, we have

$$\begin{aligned}
\mathbb{E}\left(X_2^2\right) &= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_{2_L}\sqrt{\left(1-\rho_L^2\right)}} \int_0^\infty x_2 e^{-\frac{\left(\ln x_2 - \mu_{x_2|x_1}\right)^2}{2\sigma_{2_L}^2\left(1-\rho_L^2\right)}} dx_2 \right) dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \left(e^{2\mu_{x_2|x_1}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{2\sigma_{2_L}\sqrt{\left(1-\rho_L^2\right)}w_2} e^{-\frac{w_2^2}{2}} dw_2 \right) dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_1} \int_0^\infty \frac{1}{x_1} e^{2\mu_{x_2|x_1} + 2\sigma_{2_L}^2\left(1-\rho_L^2\right)} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} dx_1, \\
&= e^{2\mu_{2_L} + 2\sigma_{2_L}^2\left(1-\rho_L^2\right)} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{2\rho\sigma_{2_L} w_1} e^{-\frac{w_1}{2}} dw_1, \\
&= e^{2\mu_{2_L} + 2\sigma_{2_L}^2\left(1-\rho_L^2\right) + \rho_L^2\sigma_{2_L}^2}, \\
&= e^{2\mu_{2_L} + 2\sigma_{2_L}^2}.
\end{aligned}$$

Therefore, the variances of the two lognormal random variables are those of their associated marginal lognormal distributions. Finally, we consider the covariance between two lognormal variables. First, we require $\mathbb{E}[X_1, X_2]$, which is

$$\begin{aligned}
\mathbb{E}(X_1 X_2) &= \frac{1}{\sqrt{2\pi}\sigma_{1_L}} \int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_L}}{\sigma_{1_L}^2}\right)^2} \left(\frac{1}{\sqrt{2\pi}\sigma_2\sqrt{\left(1-\rho_L^2\right)}} \int_0^\infty e^{-\frac{\left(\ln x_2 - \mu_{x_2|x_1}\right)^2}{2\sigma_{2_L}^2\left(1-\rho_L^2\right)}} dx_2 \right) dx_1, \\
&= e^{\mu_{2_L} + \mu_{1_L} + \frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^\infty e^{\left(\rho_L\sigma_{2_L} + \sigma_{1_L}\right)w_1} e^{-\frac{w_1^2}{2}} dw_2, \\
&= e^{\mu_{2_L} + \mu_{1_L} + \frac{\sigma_{2_L}^2\left(1-\rho_L^2\right)}{2}} e^{\frac{\left(\rho_L\sigma_{2_L} + \sigma_{1_L}\right)^2}{2}}, \\
&= e^{\mu_{2_L} + \mu_{1_L} + \frac{\sigma_{2_L}^2}{2} + \frac{\sigma_{1_L}^2}{2} + \rho_L\sigma_{1_L}\sigma_{2_L}}.
\end{aligned} \tag{4.47}$$

Combining (4.47) with the product of (4.46) and (4.45) results in covariance of two lognormal random variables as

$$\begin{aligned}
Cov[X_1, X_2] &= \mathbb{E}[X_1 X_2] - \mathbb{E}[X_1]\mathbb{E}[X_2], \\
&= e^{\mu_{2_L} + \mu_{1_L} + \frac{\sigma_{2_L}^2}{2} + \frac{\sigma_{1_L}^2}{2} + \rho_L\sigma_{1_L}\sigma_{2_L}} - e^{\mu_{1_L} + \frac{\sigma_{1_L}^2}{2}} e^{\mu_{2_L} + \frac{\sigma_{2_L}^2}{2}},
\end{aligned}$$

$$
\begin{aligned}
&= e^{\mu_{1_L} + \frac{\sigma_{1_L}^2}{2}} \left(e^{\rho_L \sigma_{1_L} \sigma_{2_L}} - 1\right) e^{\mu_{2_L} + \frac{\sigma_{2_L}^2}{2}}, \\
&= \mathbb{E}[X_1] \left(e^{\rho_L \sigma_{1_L} \sigma_{2_L}} - 1\right) \mathbb{E}[X_2].
\end{aligned} \tag{4.48}
$$

4.3.3 Median of the Bivariate Lognormal Distribution

As we mentioned for the bivariate and multivariate Gaussian distributions, the median of a multivariate distribution is a non-unique descriptive statistic; however, the most commonly defined median for the multivariate lognormal distribution is derived through applying the change of variable from the lognormal random variable to their equivalent Gaussian random variables $\hat{x}_1 \equiv \ln x_1$ and $\hat{x_2} \equiv \ln x_2$. Therefore, the Gaussian medians for $\hat{x}_1$ and $\hat{x}_2$ are μ_1 and μ_2.

The next step is to invert the medians for $\hat{\mathbf{x}}$ to $\mathbf{x}$, which results in the expression for the median as

$$
\mathbf{x}_{med} = e^{\boldsymbol{\mu}}. \tag{4.49}
$$

4.3.4 Maximum Likelihood State of a Bivariate Lognormal Distribution

The maximum likelihood state is usually found through differentiating the definition of the probability density function; however, for the multivariate forms of the lognormal distributions we have the product term scaling the exponential, $\prod_{i=1}^{N} \frac{1}{x_i}$, which would imply that we may have to differentiate with respect to each component. To overcome this hindrance, and to be able to apply vector differentiation, we shall consider the dual problem, which is to minimize the negative logarithm of (4.42). This then implies that we are seeking the minimum of

$$
J(\mathbf{x}) = \frac{1}{2} (\ln \mathbf{x} - \boldsymbol{\mu}_L)^T \boldsymbol{\Sigma}_L^{-1} (\ln \mathbf{x} - \boldsymbol{\mu}_L) + \langle \ln \mathbf{x}, \mathbf{1} \rangle, \tag{4.50}
$$

where

$$
\langle \ln \mathbf{x}, \mathbf{1} \rangle \equiv \sum_{i=1}^{N} \ln x_i,
$$

where $\mathbf{1}$ is a column vector of 1s of length N. **Note:** We have ignored the logarithm of the constant factors at the beginning of the distribution, as they do not play a factor in the derivation of the mode.

Therefore, differentiating (4.50) with respect to $\mathbf{x}$ and setting to zero results in

$$
\left.\frac{dJ}{d\mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{mode}} = \mathbf{W}^T \boldsymbol{\Sigma}^{-1} (\ln \mathbf{x}_{mode} - \boldsymbol{\mu}_L) + \langle \mathbf{W}, \mathbf{1} \rangle = \mathbf{0}, \tag{4.51}
$$

where

$$
\mathbf{W} \equiv \begin{pmatrix} x_1^{-1} & & & \\ & x_2^{-1} & & \\ & & \ddots & \\ & & & x_N^{-1} \end{pmatrix}.
$$

Factorizing the $\mathbf{W}$ term so that the remaining terms must equal $\mathbf{0}$ results in the expression for the mode of the multivariate lognormal distribution as

$$\mathbf{x}_{mode} = e^{\boldsymbol{\mu}_L - \langle \boldsymbol{\Sigma}_L, \mathbf{1} \rangle}. \tag{4.52}$$

Remark 4.5. An important feature about the mode of the bivariate lognormal distribution is that it is a function of the covariance between X_1 and X_2 through the row sum of the covariance matrix, which is what the inner product represents in (4.52). The exponential in (4.52) is applied componentwise for x_1 and x_2.

Therefore, to summarize, and to emphasize an important identity for the three descriptive statistics of the bivariate lognormal distribution, we have

$$e^{\boldsymbol{\mu}} - \langle \boldsymbol{\Sigma}, \mathbf{1} \rangle < e^{\boldsymbol{\mu}} < e^{\boldsymbol{\mu} + \frac{1}{2} \langle diag(\boldsymbol{\Sigma}), \mathbf{1} \rangle},$$

which means that the mode is always less than the median which is less than the mean. This is an important property that we need to remember for the derivation of non-Gaussian-based variational data assimilation schemes later.

4.3.5 Multivariate Lognormal Distribution

We now move on to the multivariate lognormal distribution where, given a vector, $\mathbf{x}$ of N random variables, where $x_i \in (0, \infty)$ for $i = 1, 2, \ldots, N$, the associated multivariate lognormal distribution is defined as

$$MLN(\boldsymbol{\mu}_L, \boldsymbol{\Sigma}_L) \equiv \frac{1}{(2\pi)^{\frac{N}{2}} |\boldsymbol{\Sigma}_L|^{\frac{1}{2}}} \prod_{i=1}^{N} \left(\frac{1}{x_i} \right) e^{-\frac{1}{2} (\ln \mathbf{x} - \boldsymbol{\mu}_L)^T \boldsymbol{\Sigma}_L^{-1} (\ln \mathbf{x} - \boldsymbol{\mu}_L)}, \tag{4.53}$$

where the covariance matrix for the multivariate lognormal distribution is defined by

$$\begin{aligned} \boldsymbol{\Sigma} &\equiv \begin{pmatrix} Var[\ln X_1] & Cov[\ln X_1, \ln X_2] & Cov[\ln X_1, \ln X_3] & \cdots & Cov[\ln X_1, \ln X_N] \\ Cov[\ln X_2, \ln X_1] & Var[\ln X_2] & Cov[\ln X_2, \ln X_3] & \cdots & Cov[\ln X_2, \ln X_N] \\ Cov[\ln X_3, \ln X_1] & Cov[\ln X_3, \ln X_2] & Var[\ln X_3] & \ddots & Cov[\ln X_3, \ln X_N] \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ Cov[\ln X_N, \ln X_1] & Cov[\ln X_N, \ln X_2] & Cov[\ln X_N, \ln X_3] & \cdots & Var[\ln X_N] \end{pmatrix}, \\ &\equiv \begin{pmatrix} \sigma_1^2 & \rho_{(1,2)_L} \sigma_{1_L} \sigma_{2_L} & \rho_{(1,3)_L} \sigma_{1_L} \sigma_{3_L} & \cdots & \rho_{(1,N)_L} \sigma_{1_L} \sigma_{N_L} \\ \rho_{(2,1)_L} \sigma_{2_L} \sigma_{1_L} & \sigma_{2_L}^2 & \rho_{(2,3)_L} \sigma_{2_L} \sigma_{3_L} & \cdots & \rho_{(2,N)_L} \sigma_{2_L} \sigma_{N_L} \\ \rho_{(3,1)_L} \sigma_{3_L} \sigma_{1_L} & \rho_{(3,2)_L} \sigma_{3_L} \sigma_{2_L} & \sigma_{3_L}^2 & \ddots & \rho_{(3,N)_L} \sigma_{3_L} \sigma_{N_L} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{(N,1)_L} \sigma_{N_L} \sigma_{1_L} & \rho_{(N,2)_L} \sigma_{N_L} \sigma_{2_L} & \rho_{(N,3)_L} \sigma_{N_L} \sigma_{3_L} & \cdots & \sigma_{N_L}^2 \end{pmatrix}, \end{aligned} \tag{4.54}$$

where $\rho_{(i,j)_L}$ is the correlation coefficient between the Gaussian random variables $\ln x_i$ and $\ln x_j$.

As we showed for the multivariate Gaussian distribution, it is possible to factorize the vector of means and the covariance matrix to obtain multivariate lognormal marginal and conditional distributions. The important thing to remember when deriving the conditional and marginal distributions of the multivariate lognormal distribution is to factorize the $\prod_{i=1}^{N} \frac{1}{x_i}$ term correctly, to ensure we maintain a multivariate lognormal distribution for those specific lognormal random variables.

All three of the descriptive statistics for the multivariate lognormal distribution can easily be derived from the derivation for their bivariate versions. However, we should note here—and we shall emphasize this again in the non-Gaussian data assimilation chapter—that the vector of the means for a set of lognormally distributed random variables is a componentwise expression. This means that each component is only a function of its mean and its variance. The most common median for the multivariate lognormal distribution $e^{\boldsymbol{\mu}}$ has the property that it is invariant; that is to say that it is neither a function of the variances nor the covariances, of the multivariate lognormal distribution. Thus the mode remains, which is a function of the row sums of the covariance matrix, and so this descriptive statistic is the only one that is a function of the covariances between $\ln x_i$ and the remaining random variables $\ln x_j$ for $j \neq i$.

4.4 Mixed Gaussian-Lognormal Distribution

The distribution that we introduce in this section has been referred to by three different names in its development. In its original development and in the first paper that it was presented in but rejected, it was named after the two creators, Drs. Steven Fletcher and Milija Zupanski, as the Fletcher-Zupanski distribution. After the rejection, but upon its final acceptance, it became known as the *hybrid normal-lognormal distribution* [136] and was also accepted with that name in [137]. However, with its use in [129], there was push-back about using the term *hybrid*, as at that time the development of the hybrid data assimilation systems was becoming more prevalent. Thus, to avoid confusion with that development it arrived at the name it is known by now: the **mixed Gaussian-lognormal distribution**.

The motivation for the development of this distribution came about because of the implementation of the lognormal-based variational theory from [135] into the Maximum Likelihood Ensemble Filter (MLEF) [507]. It was assumed that we could not assimilate the lognormal distributed observations at the same time as the Gaussian distribution; Dr. Fletcher did not believe this to be true, because it would mean we would have to assimilate observations sequentially rather than simultaneously; by this it is meant that we would have to run two different distribution-based assimilation systems one after the other.

Therefore, given this motivation, we introduce the bivariate form of the mixed Gaussian-lognormal distribution. As the name suggests, the associated PDF here combines features from the univariate Gaussian and lognormal distributions, and is defined by

$$MX_{1,1}(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{(1-\rho_{mx}^2)}x_1} e^{-\frac{1}{(1-\rho_{mx}^2)}\frac{Q}{2}}, \tag{4.55}$$

where

$$Q = \left(\frac{\ln x_1 - \mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2 - 2\rho_{mx}\left(\frac{\ln x_1 - \mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)\left(\frac{x_2 - \mu_{2_{mx}}}{\sigma_{2_{mx}}}\right) + \left(\frac{x_2 - \mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2. \tag{4.56}$$

The correlation term ρ_{mx} is defined as

$$\rho_{mx} \equiv \frac{Cov\left[\ln X_1, X_2\right]}{\sqrt{Var\left[\ln X_1\right] Var\left[X_2\right]}}, \tag{4.57}$$

which therefore contains the covariance between the lognormal random variable and the Gaussian random variable, for $x_1 \in (0, \infty)$ and $x_2 \in (-\infty, \infty)$.

For us to be able to derive the moments of the mixed bivariate Gaussian-lognormal distribution, we need to be able to express (4.55) combined with (4.56) as the product of a marginal and conditional distribution. However, the question is: with respect to which distribution type?

The first factorization we show is for a marginal Gaussian distribution and a conditional lognormal distribution, which is given by

$$MX_{1,1}\left(x_1, x_2\right) = \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}} e^{-\frac{1}{2}\left(\frac{x_2 - \mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2} \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}\sqrt{\left(1 - \rho_{mx}^2\right)}} \frac{1}{x_1} e^{-\frac{\left(\ln x_1 - \mu_{x_1|x_2}\right)^2}{2\sigma_{1_{mx}}^2\left(1 - \rho_{mx}^2\right)}}, \tag{4.58}$$

where for this distribution the lognormal conditional mean, $\mu_{x_1|x_2}$ is

$$\mu_{x_1|x_2} = \mu_{1_{mx}} + \rho_{mx}\frac{\sigma_{1_{mx}}}{\sigma_{2_{mx}}}\left(x_2 - \mu_{2_{mx}}\right). \tag{4.59}$$

If we were to factorize (4.55) into the product of a marginal lognormal distribution and a conditional Gaussian distribution, then we would have

$$MX_{1,1}\left(x_1, x_2\right) = \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}} \frac{1}{x_1} e^{-\frac{1}{2}\left(\frac{\ln x_1 - \mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2} \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}\sqrt{\left(1 - \rho_{mx}^2\right)}} e^{-\frac{\left(x_2 - \mu_{x_2|x_1}\right)^2}{2\sigma_{2_{mx}}^2\left(1 - \rho_{mx}^2\right)}}, \tag{4.60}$$

where the Gaussian conditional mean is given by

$$\mu_{x_2|x_1} = \mu_{2_{mx}} + \rho_{mx}\frac{\sigma_{2_{mx}}}{\sigma_{1_{mx}}}\left(\ln x_1 - \mu_{1_{mx}}\right). \tag{4.61}$$

4.4.1 Moments of the Bivariate Mixed Gaussian-Lognormal Distribution

We now consider the properties of the distribution. The first property that we consider involves the moments of the marginal distribution. We start with the first moments (means) of (4.60). Therefore,

$\mathbb{E}[X_1]$ is

$$\begin{aligned}\mathbb{E}[X_1] &= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_{-\infty}^\infty e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_2\right)dx_1,\\ &= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^\infty e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}dx_1,\\ &= e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2}{2}},\end{aligned} \tag{4.62}$$

which is the mean of a univariate lognormal distribution.

The expectation of x_2 from (4.58) gives us

$$\begin{aligned}\mathbb{E}[X_2] &= \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}}\int_{-\infty}^\infty x_2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_0^\infty \frac{1}{x_1}e^{-\frac{\left(\ln x_1-\mu_{x_1|x_2}\right)^2}{2\sigma_{1_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_1\right)dx_2,\\ &= \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}}\int_{-\infty}^\infty x_2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2}dx_2,\\ &= \mu_{2_{mx}},\end{aligned} \tag{4.63}$$

which is of course the mean of a univariate Gaussian distribution.

It is possible to obtain (4.62) and (4.63) by using the definition for the mixed distribution from (4.60), but this is left as an exercise.

Exercise 4.6. *Verify that the first-order moments for the factorization in* (4.60) *are equal to* (4.62) *and* (4.63).

The second order non-central moment for x_1, given the factorization in (4.60), is

$$\begin{aligned}\mathbb{E}\left[X_1^2\right] &= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^\infty x_1 e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_{-\infty}^\infty e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_2\right)dx_1,\\ &= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^\infty x_1 e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}dx_1,\\ &= e^{2\mu_{1_{mx}}+2\sigma_{1_{mx}}^2},\end{aligned} \tag{4.64}$$

which is again the same as the univariate lognormal distribution.

The second order non-central moment for x_2, is calculated by

$$\begin{aligned}
\mathbb{E}\left[X_2^2\right] &= \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}}\int_{-\infty}^{\infty} x_2^2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_0^{\infty}\frac{1}{x_1}e^{-\frac{\left(\ln x_1-\mu_{x_1|x_2}\right)^2}{2\sigma_{1_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_1\right)dx_2, \\
&= \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}}\int_{-\infty}^{\infty} x_2^2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2}dx_2, \\
&= \mu_{2_{mx}}^2+\sigma_{2_{mx}}^2. \qquad (4.65)
\end{aligned}$$

Considering the other factorization, (4.60), we have

$$\begin{aligned}
\mathbb{E}\left[X_2^2\right] &= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^{\infty}\frac{1}{x_1}e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_1}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_{-\infty}^{\infty} x_2^2 e^{-\frac{\left(x_2-\mu_{x_2|x_1}\right)^2}{2\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_2\right)dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}}\int_0^{\infty}\frac{1}{x_1}e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}\left(\mu_{x_1|x_2}^2+\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)\right)dx_1, \\
&= \frac{1}{\sqrt{2\pi}\sigma_1}\int_0^{\infty}\frac{1}{x_1}\left(\mu_{2_{mx}}^2+2\mu_{2_{mx}}\rho_{mx}\frac{\sigma_{2_{mx}}}{\sigma_{1_{mx}}}\left(\ln x_1-\mu_{1_{mx}}\right)+\rho_{mx}^2\frac{\sigma_{2_{mx}}^2}{\sigma_{1_{mx}}^2}\left(\ln x_1-\mu_{1_{mx}}\right)^2\right. \\
&\quad \left.+\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)\right)\times e^{-\frac{1}{2}\left(\frac{\ln x_1-\mu_{1_{mx}}}{\sigma_{1_{mx}}}\right)^2}dx_1, \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\left(\mu_{2_{mx}}^2+\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)+2\mu_{2_{mx}}\rho_{mx}\sigma_{2_{mx}}w_1+\rho_{mx}^2\sigma_{2_{mx}}^2w_1^2\right)e^{-\frac{w_1^2}{2}}dw_1, \\
&= \mu_{2_{mx}}^2+\sigma_{2_{mx}}^2\left(1-\rho_{mx}^2\right)+\rho_{mx}^2\sigma_{2_{mx}}^2=\mu_{2_{mx}}^2+\sigma_{2_{mx}}^2, \qquad (4.66)
\end{aligned}$$

which is the definition of the second order non-central moment of a univariate Gaussian distribution.

Finally, we consider the definition for the covariance for the mixed distribution, where we require $\mathbb{E}[X_1X_2]$, which is defined as

$$\mathbb{E}[X_1X_2] = \frac{1}{\sqrt{2\pi}\sigma_{2_{mx}}}\int_{-\infty}^{\infty} x_2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_2}\right)^2}\left(\frac{1}{\sqrt{2\pi}\sigma_{1_{mx}}\sqrt{\left(1-\rho_{mx}^2\right)}}\int_0^{\infty} e^{-\frac{\left(\ln x_1-\mu_{x_1|x_2}\right)^2}{2\sigma_{1_{mx}}^2\left(1-\rho_{mx}^2\right)}}dx_1\right)dx_2,$$

$$
\begin{aligned}
&= \frac{1}{\sqrt{2\pi}\sigma_2}\int_{-\infty}^{\infty} x_2 e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_2}\right)^2} e^{\mu_{x_1|x_2}+\frac{\sigma_1^2\left(1-\rho_{mx}^2\right)}{2}} dx_2,\\
&= e^{\mu_{1_{mx}}+\frac{\sigma_1^2\left(1-\rho_{mx}^2\right)}{2}} \frac{1}{\sqrt{2\pi}\sigma_2}\int_{-\infty}^{\infty} x_2 e^{\rho_{mx}\frac{\sigma_{1_{mx}}}{\sigma_{2_{mx}}}\left(x_2-\mu_{2_{mx}}\right)} e^{-\frac{1}{2}\left(\frac{x_2-\mu_{2_{mx}}}{\sigma_{2_{mx}}}\right)^2} dx_2,\\
&= e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2\left(1-\rho_{mx}^2\right)}{2}} \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \left(\sigma_{2_{mx}} z_2+\mu_{2_{mx}}\right) e^{\rho_{mx}\sigma_{1_{mx}} z_2} e^{-\frac{z_2^2}{2}} dz_2,\\
&= e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2\left(1-\rho_{mx}^2\right)}{2}} e^{\frac{\rho_{mx}^2\sigma_{1_{mx}}^2}{2}} \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \left(\sigma_{2_{mx}} z_2+\mu_{2_{mx}}\right) e^{-\frac{\left(z_2-\rho_{mx}\sigma_{1_{mx}}\right)^2}{2}} dz_2,\\
&= e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2}{2}} \left(\rho_{mx}\sigma_{1_{mx}}\sigma_{2_{mx}}+\mu_{2_{mx}}\right).
\end{aligned}
\tag{4.67}
$$

Substituting (4.67) into the definition for the covariance gives us

$$
Cov\left[X_1, X_2\right] = e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2}{2}} \left(\rho_{mx}\sigma_{1_{mx}}\sigma_{2_{mx}}+\mu_{2_{mx}}\right) - \mu_{2_{mx}} e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2}{2}} = \rho_{mx}\sigma_{1_{mx}}\sigma_{2_{mx}} e^{\mu_{1_{mx}}+\frac{\sigma_{1_{mx}}^2}{2}},
\tag{4.68}
$$

which is a combination of the covariance from both the univariate Gaussian and lognormal distributions.

4.4.2 Median of the Mixed Gaussian-Lognormal Distribution

As we have seen for the bivariate and multivariate Gaussian and lognormal distributions, the median of the mixed Gaussian-lognormal distribution is calculated by integrating, as with the other two distributions. As with the lognormal distribution, we apply the logarithmic transform to the lognormal distributed variable and find the median of the equivalent bivariate Gaussian distribution and then invert back to the mixed distribution. Thus the most common median of the mixed Gaussian-lognormal distribution is

$$
x_{med} = \begin{pmatrix} e^{\mu_{1_{mx}}} \\ \mu_{2_{mx}} \end{pmatrix},
\tag{4.69}
$$

which we see is a combination of the medians of a univariate Gaussian distribution and a univariate lognormal distribution.

4.4.3 Maximum Likelihood Estimate for the Mixed Gaussian and Lognormal Distribution

As with the bivariate lognormal, because of the x^{-1} factor in the definition of the mixed PDF we consider the log-likelihood to find the mode, but first we rewrite the bivariate mixed Gaussian-lognormal

distribution in vectorial form, which is given by

$$MX\left(\boldsymbol{\mu}_{mx}, \boldsymbol{\Sigma}_{mx}\right) \equiv \frac{1}{2\pi\left|\boldsymbol{\Sigma}_{mx}\right|} \frac{1}{x_1} e^{-\frac{1}{2}\begin{pmatrix} \ln x_1 - \mu_{1_{mx}} \\ x_2 - \mu_{2_{mx}} \end{pmatrix}^T \boldsymbol{\Sigma}_{mx}^{-1} \begin{pmatrix} \ln x_1 - \mu_{1_{mx}} \\ x_2 - \mu_{2_{mx}} \end{pmatrix}}, \tag{4.70}$$

and therefore the negative logarithm of (4.70) results in

$$J\left(\mathbf{x}\right) = \frac{1}{2}\begin{pmatrix} \ln x_1 - \mu_{1_{mx}} \\ x_2 - \mu_{2_{mx}} \end{pmatrix}^T \boldsymbol{\Sigma}_{mx}^{-1} \begin{pmatrix} \ln x_1 - \mu_{1_{mx}} \\ x_2 - \mu_{2_{mx}} \end{pmatrix} + \left\langle \begin{pmatrix} \ln x_1 \\ x_2 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\rangle. \tag{4.71}$$

Differentiating (4.71) and setting to zero results in the mode of the bivariate Gaussian-lognormal distribution, which can easily be shown to be

$$\mathbf{x}_{mode} = \begin{pmatrix} e^{\mu_{1_{mx}} - \sigma_{1_{mx}}^2} \\ \mu_{2_{mx}} + \rho_{mx}\sigma_{1_{mx}}\sigma_{2_{mx}} \end{pmatrix}. \tag{4.72}$$

The first feature to note in (4.72) is that the component of the mode related to the lognormal random variable stays the same as for the univariate lognormal distribution. However, the mode for the Gaussian component is **not** as that for the univariate Gaussian distribution. It is now a **function of the covariances between the Gaussian and the lognormal random variables**. Therefore, the mode of the mixed Gaussian-lognormal distribution is a **function of the covariance between the Gaussian and lognormal random variables**.

4.4.4 Diagrams of the Bivariate Gaussian-Lognormal Distribution

In this subsection we present plots of the bivariate Gaussian-lognormal distribution. In Fig. 4.1 we have plotted 2D and 3D contour plots of the bivariate Gaussian, lognormal, and mixed distributions. In all of the plots in Fig. 4.1 we have $\rho_{mx} = 0$ (uncorrelated case), but with different values for the variances of the two random variables. The mode of the uncorrelated mixed distribution is situated at the coordinates of the mode of the univariate lognormal and Gaussian distributions. For the Gaussian distribution the mode is $\mu_{2_{mx}}$ and for the lognormal this is $e^{\mu_{1_{mx}} - \sigma_{1_{mx}}^2}$. We can see that the mixed distribution inherits the circular structure similar to the bivariate Gaussian distribution, but also has the tails associated with the bivariate lognormal distribution. The advantage of the mixed distribution is that if we had an error that appears to be an outlier with the bivariate Gaussian distribution, it would be captured by either the bivariate lognormal or the mixed distributions.

In Fig. 4.2 we have introduced a positive correlation of $\rho = 0.5$, but with the same values for the variance in Fig. 4.1. The effect that this choice for the correlation coefficient has is to stretch the bivariate Gaussian distribution, but retains its symmetric properties about the mean/median/mode. However, we see that both the lognormal and the mixed distributions show signs of skewness. We should note that the mixed distribution keeps the circular structure associated with the Gaussian distribution; however, it does appear to have a larger skewness compared to the other two distributions.

The next four figures, Figs. 4.3–4.6, show the effect different standard deviations for the lognormal variable and the Gaussian variables have on the bivariate mixed distribution. The main feature is that when the lognormal standard deviation increases, the bivariate mixed distribution mode moves toward

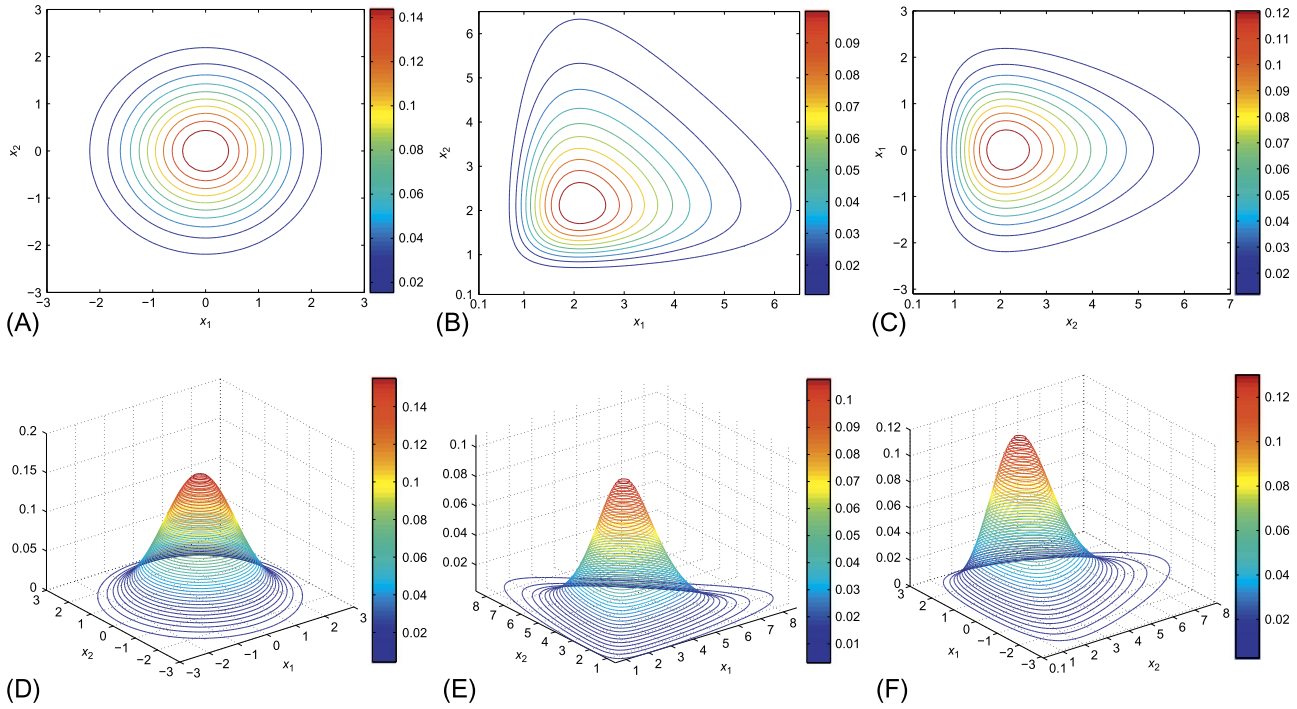


FIGURE 4.1

(A) Contour plot of the unit uncorrelated bivariate Gaussian distribution's PDF; (B) Contour plot of the uncorrelated bivariate lognormal distribution's PDF with $\mu_1 = \mu_2 = 1$, and $\sigma_1 = \sigma_2 = 0.5$; (C) Contour plot of the uncorrelated mixed distribution PDF with unit Gaussian random variable and lognormal random variable with $\mu_1 = 1$, $\sigma_1 = 0.5$; (D) 3D contour plot of (A); (E) 3D contour plot of (B); (F) 3D contour plot of (C).

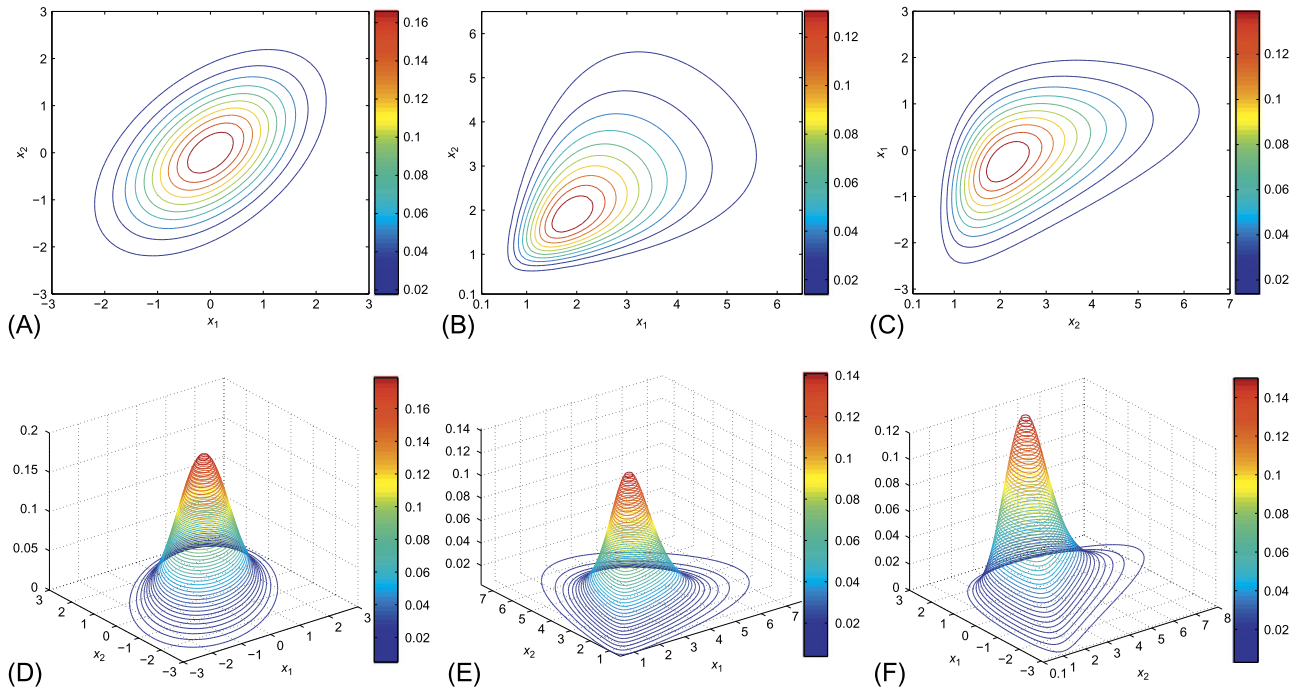


FIGURE 4.2

(A) Contour plot of the 0.5 positive correlated bivariate Gaussian distribution's PDF; (B) Contour plot of the 0.5 positive correlated bivariate lognormal distribution's PDF with $\mu_1 = \mu_2 = 1$, $\sigma_1 = \sigma_2 = 0.5$; (C) Contour plot of the 0.5 positive correlated $FZ_{1,1}$'s PDF with unit Gaussian random variable and lognormal random variable with $\mu_l = 1$, $\sigma_l = 0.5$; (D) 3D contour plot of (A); (E) 3D contour plot of (B); (F) 3D contour plot of (C).

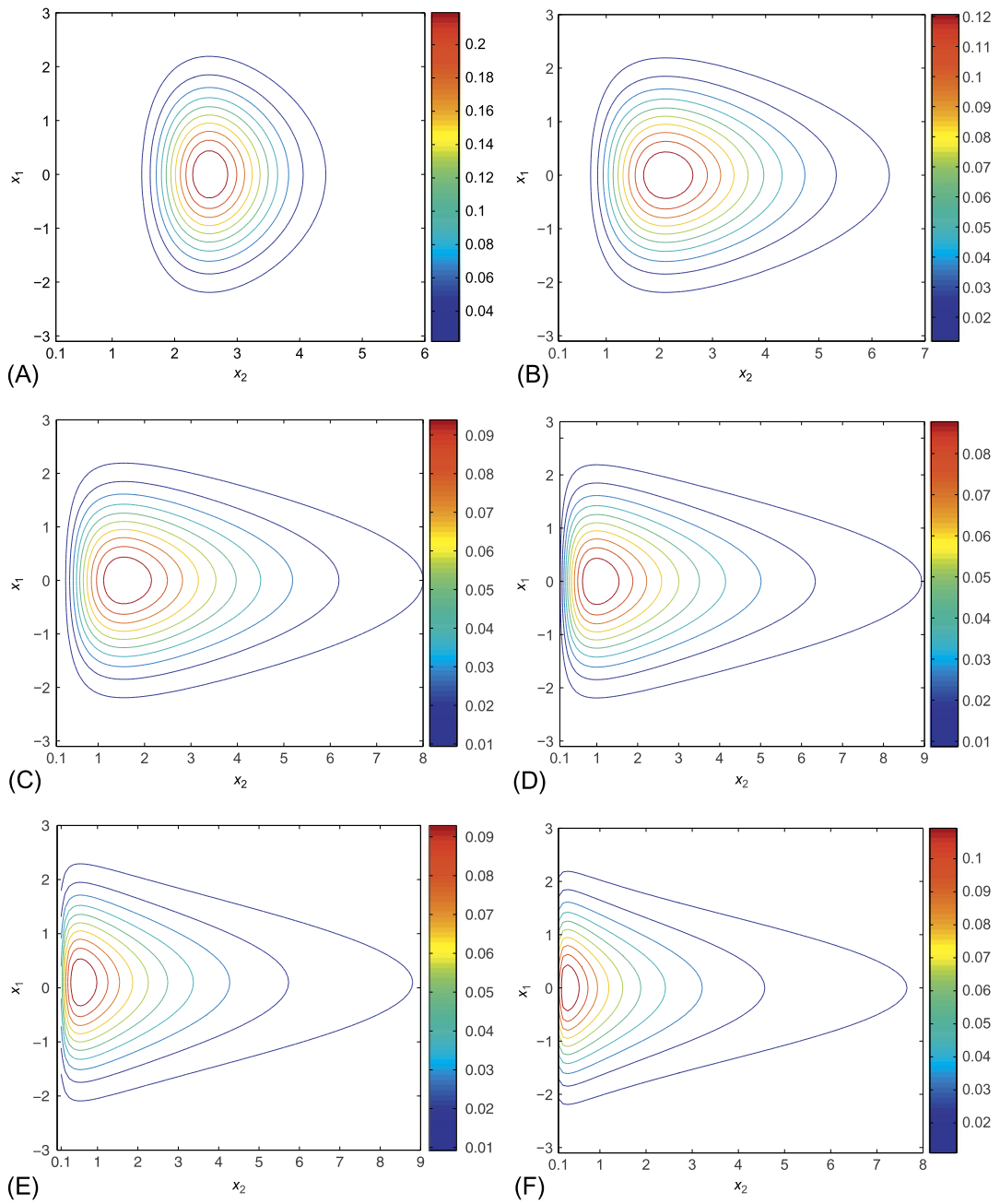


FIGURE 4.3

(A) Contour plot with $\sigma_l = 0.25$; (B) Contour plot with $\sigma_l = 0.5$, (C) Contour plot with $\sigma_l = 0.75$, (D) Contour plot with $\sigma_l = 1$; (E) Contour plot with $\sigma_l = 1.25$; (F) Contour plot with $\sigma_l = 1.5$; (G) Contour plot with $\sigma_l = 1.75$; (H) Contour plot with $\sigma_l = 2$.

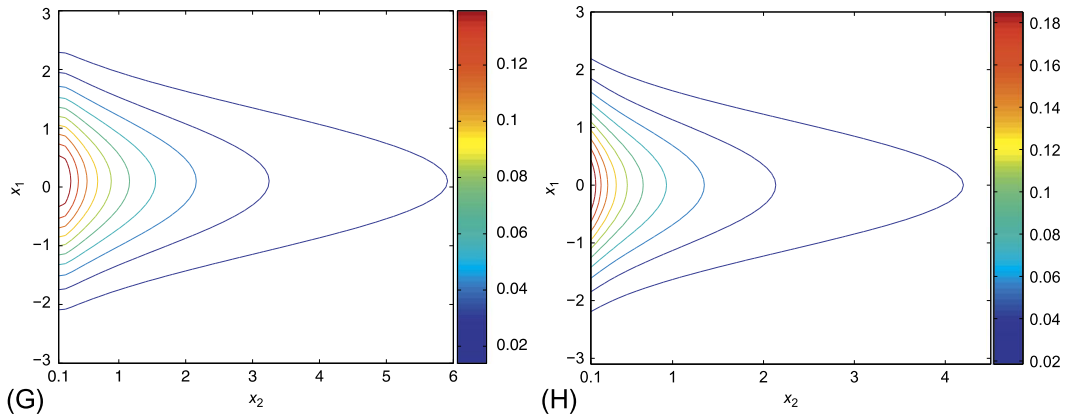


FIGURE 4.3

(*continued*)

the x_2 axis due to the definition of the mode. If $\mu_{1_{mx}} > \sigma^2_{1_{mx}}$, then the tail of the bivariate mixed distribution stretches along the x_1 axis. When the Gaussian standard deviation increases and the lognormal standard deviation is constant, the distribution stretches along the x_2 axis. When both standard deviations change, as in Figs. 4.5 and 4.6, we see that for $\rho_{mx} = 0$ the mode moves toward the x_2 axis as associated with the lognormal component. With the introduction of the correlation, we see that there is a skewness and a break with the symmetry that we noted with the uncorrelated mixed distribution. This again allows for what may be considered outliers in the other two distributions.

4.5 Multivariate Mixed Gaussian-Lognormal Distribution

The original motivation for introduction of the mixed distribution was to be able to use it in a variational formulation of data assimilation for numerical weather prediction. Therefore, the multivariate extension from the bivariate mixed distribution is defined by

$$MX_{p,q}\left(\boldsymbol{\mu}_{mx}, \boldsymbol{\Sigma}_{mx}\right) = \frac{1}{2\pi^{\frac{p+q}{2}} |\boldsymbol{\Sigma}_{mx}|^{\frac{1}{2}}} \prod_{i=1}^{q} \left(\frac{1}{x_i}\right) e^{-\frac{1}{2}(\widehat{\mathbf{x}}-\boldsymbol{\mu}_{mx})^T \boldsymbol{\Sigma}_{mx}^{-1}(\widehat{\mathbf{x}}-\boldsymbol{\mu}_{mx})}, \tag{4.73}$$

where p represents the total number of Gaussian random variables and q is the total number of lognormal random variables, $\boldsymbol{\mu}_{mx}$ is the vector of means with dimension $(p+q) \times 1$, $\hat{\mathbf{x}}$ is the vector of the random variables given by

$$\widehat{\mathbf{x}}^T \equiv \left(\underbrace{x_1 \quad x_2 \quad \ldots \quad x_p}_{Gaussian} \underbrace{\ln x_{p+1} \quad \ln x_{p+2} \quad \ldots \quad \ln x_k}_{lognormal} \right),$$

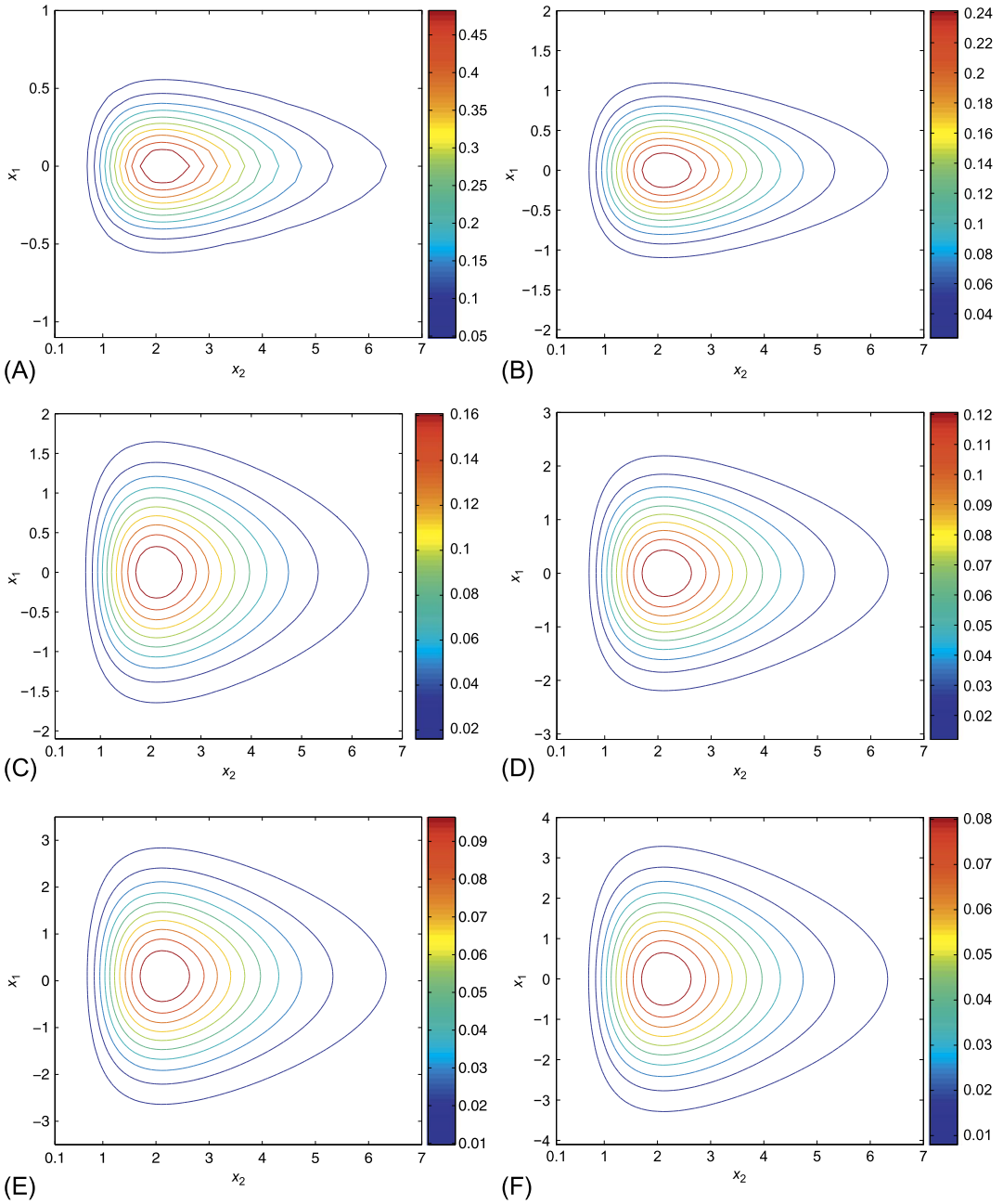


FIGURE 4.4

(A) Contour plot with $\sigma_G = 0.25$; (B) Contour plot with $\sigma_G = 0.5$; (C) Contour plot with $\sigma_G = 0.75$; (D) Contour plot with $\sigma_G = 1$; (E) Contour plot with $\sigma_G = 1.25$; (F) Contour plot with $\sigma_G = 1.5$; (G) Contour plot with $\sigma_G = 1.75$; (H) Contour plot with $\sigma_G = 2$.

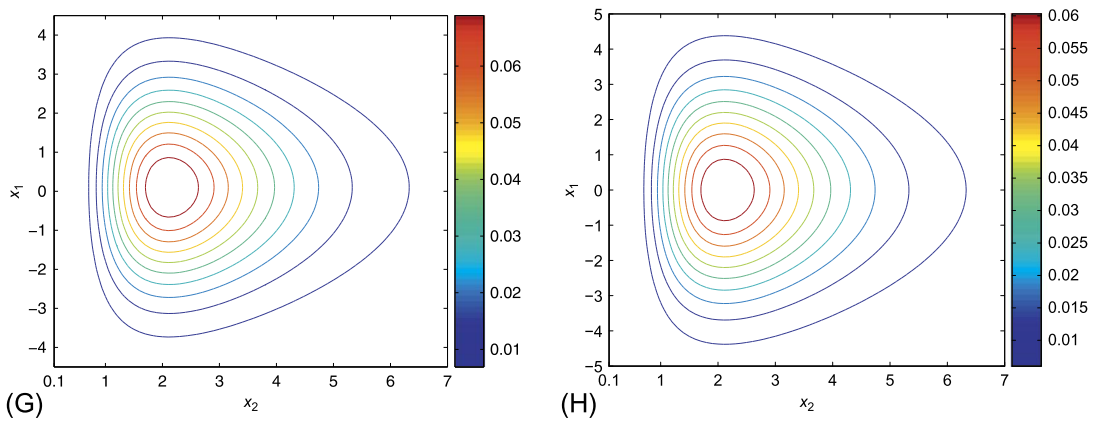


FIGURE 4.4

(*continued*)

and $\boldsymbol{\Sigma}_{mx}$ is the covariance matrix for the multivariate mixed distribution, given by

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma^2_{1_{mx}} & \rho_{(1)_{mx}}\sigma_{1_{mx}}\sigma_{2_{mx}} & \rho_{(12)_{mx}}\sigma_{1_{mx}}\sigma_{3_{mx}} & \cdots & \rho_{(1N)_{mx}}\sigma_{1_{mx}}\sigma_{N_{mx}} \\ \rho_{(21)_{mx}}\sigma_{1_{mx}}\sigma_{2_{mx}} & \sigma^2_{2_{mx}} & \rho_{(23)_{mx}}\sigma_{2_{mx}}\sigma_{3_{mx}} & \cdots & \rho_{(2n)_{mx}}\sigma_{2_{mx}}\sigma_{N_{mx}} \\ \rho_{(13)_{mx}}\sigma_{3_{mx}}\sigma_{1_{mx}} & \rho_{(32)_{mx}}\sigma_{3_{mx}}\sigma_{2_{mx}} & \sigma^2_{3_{mx}} & \cdots & \rho_{(3n)_{mx}}\sigma_{3_{mx}}\sigma_{N_{mx}} \\ \vdots & \vdots & & \ddots & \vdots \\ \rho_{(N1)_{mx}}\sigma_{N_{mx}}\sigma_{1_{mx}} & \rho_{(N2)_{mx}}\sigma_{N_{mx}}\sigma_{2_{mx}} & \rho_{(NN-1)_{mx}}\sigma_{N_{mx}}\sigma_{(N-1)_{mx}} & \cdots & \sigma^2_{N_{mx}} \end{pmatrix},$$

where $N = p + q$.

As we saw with the bivariate case, the covariance matrix for the mixed distribution couples the Gaussian and lognormal random variables when they are not independent. Therefore, this distribution enables changes in the lognormal random variables to influence the values of the Gaussian random variables and vice versa. The proof that the definition of the multivariate mixed distribution in (4.73) is a probability density function can be found in [136]. The proof there is based on transforming the lognormally distributed random variables into Gaussian random variables and then showing that the function satisfies the conditions for a multivariate probability density function.

The original proof for the Fletcher-Zupanski distribution was based upon a proof by complete induction. We showed that a statement was true for the bivariate case and then show that it is true for the sequence of multivariate combinations up to N; we then use this to prove that the distribution is true for $N + 1$. The proof was based upon being able to factorize (4.73) into the product of a lower-order marginal and conditional distributions. However, there is more than one possible combination for these factorizations of higher-order versions of the mixed distribution. To illustrate the different combinations for the marginal and conditional PDFs, we shall consider the trivariate and quadvariate versions of (4.73).

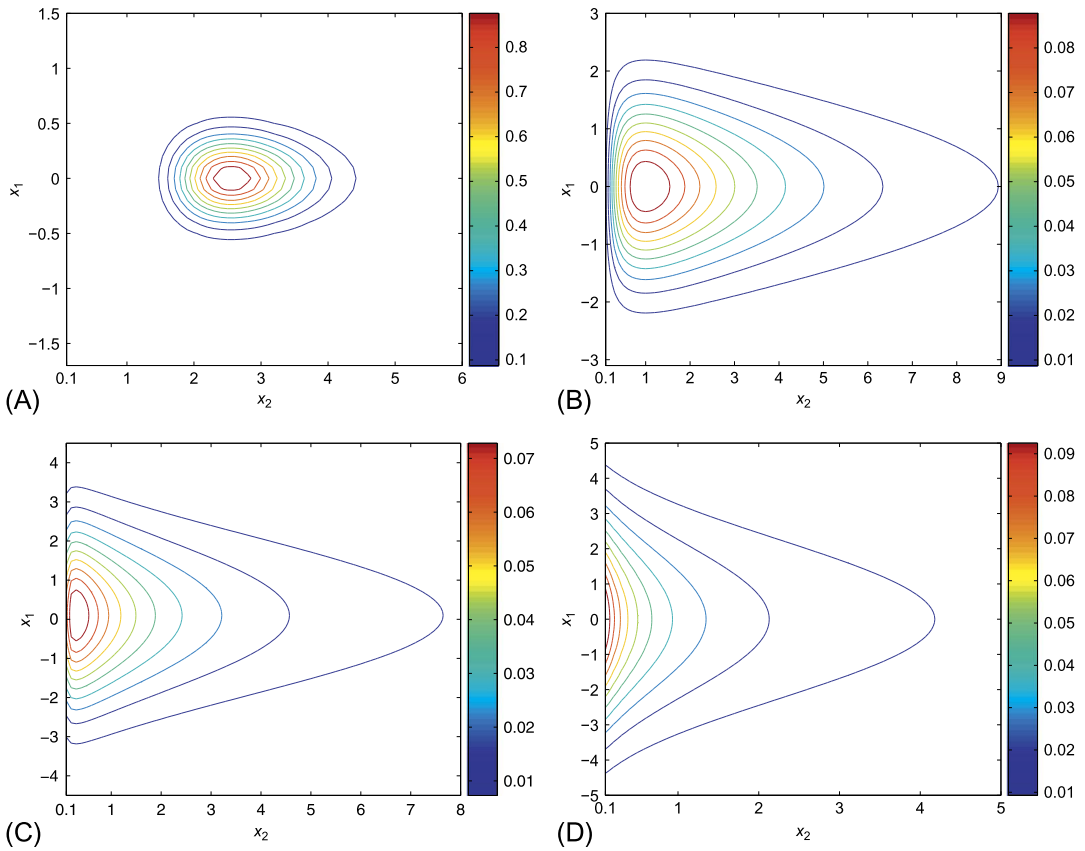


FIGURE 4.5

(A) Contour plot with $\sigma_G = \sigma_L = 0.25$; (B) Contour plot with $\sigma_G = \sigma_L = 1$; (C) Contour plot with $\sigma_G = \sigma_L = 1.5$; (D) Contour plot with $\sigma_G = \sigma_L = 2$.

4.5.1 Trivariate and Quadvariate Mixed Distribution

The first thing to note about the trivariate mixed distribution is that there are two possible combinations of Gaussian and lognormal random variables that can be described by this distribution; there are two Gaussian, one lognormal, or one Gaussian, and two lognormal random variables. A feature of the trivariate mixed distributions is that they can be factorized into the different combinations of subdistributions. We have summarized these factorizations in Table 4.1, where we see that it is possible to factorize the trivariate mixed distribution into combinations of univariate or bivariate Gaussian or lognormal distributions, as well as the bivariate mixed distribution.

We have summarized the marginal and conditional distribution combinations for the quadvariate mixed Gaussian-lognormal distribution in Table 4.2, where we see the different possible combinations of the univariate and bivariate Gaussian and lognormal distributions. We now have the mixed distribution with two or one Gaussian and lognormal random variables, or the other way around.

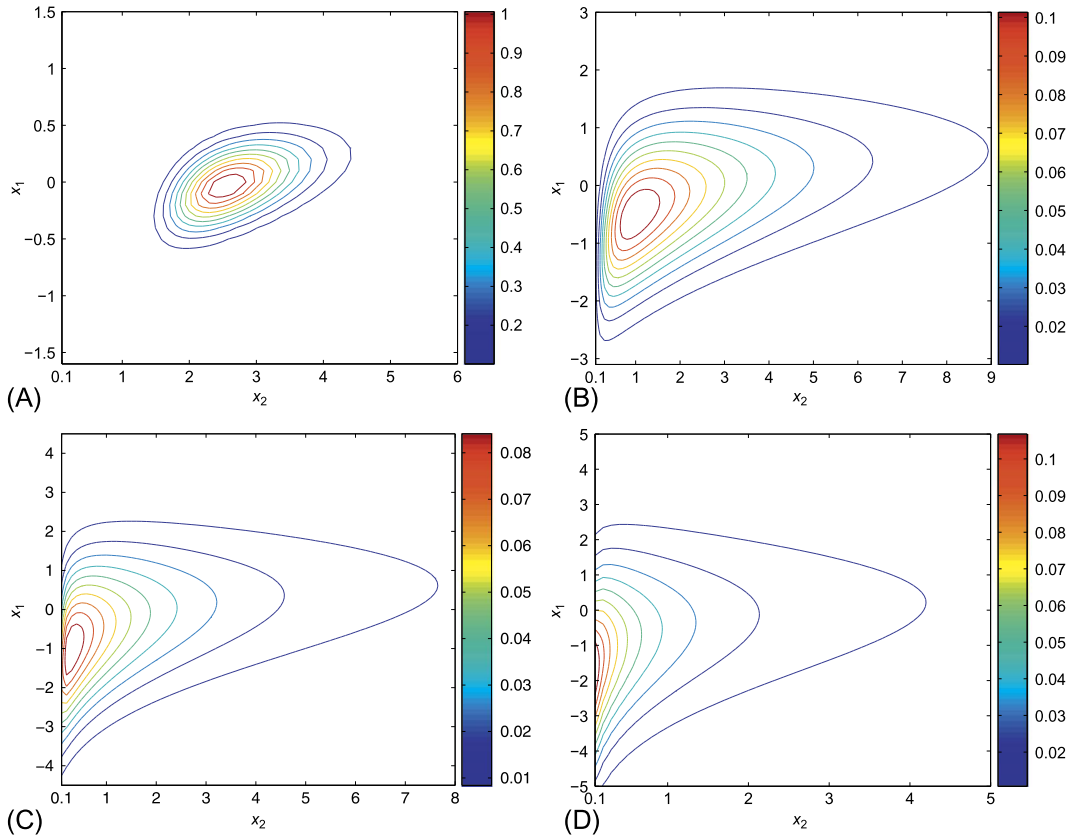


FIGURE 4.6

(A) Contour plot with $\sigma_G = \sigma_L = 0.25$; (B) Contour plot with $\sigma_G = \sigma_L = 1$; (C) Contour plot with $\sigma_G = \sigma_L = 1.5$; (D) Contour plot with $\sigma_G = \sigma_L = 2$.

Table 4.1 Different Factorizations of the Trivariate Mixed Gaussian-Lognormal Distributions.

Trivariate Mixed Distributions			
$MX_{1,2}$		$MX_{2,1}$	
Marginal	**Conditional**	**Marginal**	**Conditional**
G_1	LN_2	LN_1	G_2
LN_2	G_1	G_2	LN_1
LN_1	$MX_{1,1}$	G_1	$MX_{1,1}$
$MX_{1,1}$	LN_1	$MX_{1,1}$	G_1

Table 4.2 Different Factorizations of the Quadvariate Mixed Gaussian-Lognormal Distributions.

Quadvariate Mixed Distributions					
$MX_{2,2}$		$MX_{3,1}$		$MX_{1,3}$	
Marginal	**Conditional**	**Marginal**	**Conditional**	**Marginal**	**Conditional**
G_2	LN_2	G_3	LN_1	LN_3	G_1
LN_2	G_2	LN_1	G_3	G_1	LN_3
MX_{11}	MX_{11}	MX_{11}	G_2	MX_{11}	LN_2
		G_2	MX_{11}	LN_2	MX_{11}
		MX_{11}	G_2	MX_{11}	LN_2
		G_1	MX_{21}	LN_1	MX_{12}
		MX_{21}	G_1	MX_{12}	LN_1

4.5.2 Mode of the Multivariate Mixed Distribution

However, unlike the multivariate Gaussian and lognormal distribution, we remark here about the mixed distribution's mode for higher order than the bivariate mode. We saw for the bivariate case that the Gaussian component was a function of the covariances between the Gaussian and the lognormal random variables. For the higher-order cases, the mode is given by

$$\begin{pmatrix} \ln \mathbf{x}_q \\ \mathbf{x}_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{\mu}_q \\ \boldsymbol{\mu}_p \end{pmatrix} + \begin{pmatrix} \boldsymbol{\Sigma}_{qq} & \boldsymbol{\Sigma}_{pq} \\ \boldsymbol{\Sigma}_{qp} & \boldsymbol{\Sigma}_{pp} \end{pmatrix} \begin{pmatrix} \mathbf{1}_q \\ \mathbf{0}_p \end{pmatrix}. \tag{4.74}$$

The important feature to notice in (4.74) is that the lognormal component is equivalent to the definition for the mode of the multivariate lognormal distribution, where it is a function of the row sum of the covariances between the lognormal random variables. However, the mode associated with the Gaussian random variables is a function of the row sum of the covariances between the lognormal and the Gaussian random variables. In summary, the mode of the multivariate mixed distribution is

$$\begin{pmatrix} \mathbf{x}_q \\ \mathbf{x}_p \end{pmatrix}_{mode} = \begin{pmatrix} e^{\boldsymbol{\mu}_q + \langle \boldsymbol{\Sigma}_{qq}, \mathbf{1}_q \rangle} \\ \boldsymbol{\mu}_p + \langle \boldsymbol{\Sigma}_{qp}, \mathbf{1}_q \rangle \end{pmatrix}. \tag{4.75}$$

4.6 Reverse Lognormal Distribution

We introduced the reverse lognormal distribution as one of the two new distributions in the last chapter in this edition. We now extend this distribution into its multivariate form. We start with the bivariate version, given by

$$BR\Lambda\left(\mathbf{T} - \mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \left(\frac{1}{(T_1 - x_1)}\right)\left(\frac{1}{(T_2 - x_2)}\right)\frac{1}{2\pi\sigma_1\sigma_2\sqrt{(1-\rho^2}}$$
$$\exp\left\{-\frac{1}{2}\left(\frac{(\ln(T_1 - x_1) - \mu_1)^2}{\sigma_1^2}\right) + \left(\frac{(\ln(T_2 - x_2) - \mu_2)^2}{\sigma_2^2}\right) - 2\rho\left(\frac{(\ln(T_1 - x_1) - \mu_1)}{\sigma_1}\right)\left(\frac{(\ln(T_2 - x_2) - \mu_2)}{\sigma_2}\right)\right\} \tag{4.76}$$

for $x_1 \in (-\infty, T_1)$, and $x_2 \in (-\infty, T_2)$.

As we saw with the univariate version of this distribution, the descriptive statistics are the same as the equivalent lognormal distribution subtracted from the upper limit. Therefore, for the bivariate reverse lognormal distribution the mean, median and mode can be shown to be the following respectively:

$$\mathbf{x}_{mean} = \begin{pmatrix} T_1 - \exp\left\{\mu_1 + \dfrac{\sigma_1^2}{2}\right\} \\ T_2 - \exp\left\{\mu_2 + \dfrac{\sigma_2^2}{2}\right\} \end{pmatrix}, \quad \mathbf{x}_{median} = \begin{pmatrix} T_1 - \exp\{\mu_1\} \\ T_2 - \exp\{\mu_2\} \end{pmatrix},$$
$$\mathbf{x}_{mode} = \begin{pmatrix} T_1 - \exp\left\{\mu_1 - \sigma_1^2 - \rho\sigma_1\sigma_2\right\} \\ T_2 - \exp\left\{\mu_2 - \sigma_2^2 - \rho\sigma_1\sigma_2\right\} \end{pmatrix}. \tag{4.77}$$

4.6.1 Multivariate Reverse Lognormal Distribution

The definition of the multivariate reverse lognormal distribution is given by

$$MR\Lambda\left(\mathbf{T} - \mathbf{x}|\,\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \prod_{i=1}^{N}\left(\frac{1}{T_i - x_i}\right)\frac{1}{(2\pi)^{\frac{N}{2}}\,|\boldsymbol{\Sigma}|^{\frac{1}{2}}} \times \exp\left\{-\frac{1}{2}\left(\ln\left(\mathbf{T} - \mathbf{x}\right) - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\ln\left(\mathbf{T} - \mathbf{x}\right) - \boldsymbol{\mu}\right)\right\}, \tag{4.78}$$

where $x_i \in (-\infty, T_i)$, and $\mathbf{T}_i = T_i$, for $i = 1, 2, \ldots, N$.

As always with multivariate distributions, the mean is in component form, given by

$$x_{mean,i} = T_i - \exp\left\{\mu_i + \frac{\sigma_i^2}{2}\right\}, \quad \text{for } i = 1, 2, \ldots, N. \tag{4.79}$$

The median is given by

$$\mathbf{x}_{median} = \mathbf{T} - \exp\{\boldsymbol{\mu}\}. \tag{4.80}$$

Finally the mode is given by

$$\mathbf{x}_{mode} = \mathbf{T} - \exp\{\boldsymbol{\mu} - \langle\boldsymbol{\Sigma}, \mathbf{1}\rangle\}. \tag{4.81}$$

4.6.2 Combining With Gaussian Distribution

We showed earlier that it is possible to combine a multivariate Gaussian distribution with a lognormal distribution to obtain a mixed Gaussian-lognormal distribution and the same is true here. We start by assuming that there are p Gaussian random variables and q reverse lognormal random variables, such that $x_i \in (-\infty, \infty)$ for $i = 1, 2, \ldots, p$ and $x_i \in (-\infty, T_i)$ for $i = p+1, p+2, \ldots, N$, and $N = p + q$. Thus the definition for the mixed Gaussian-reverse-lognormal distribution is given by

$$MXGRL\left(\mathbf{x}\right) = \left(\prod_{i=p+1}^{N}\frac{1}{T_i - x_i}\right)\frac{1}{(2\pi)^{\frac{N}{2}}\,|\boldsymbol{\Sigma}|^{-\frac{1}{2}}}$$

$$\times \exp\left\{\frac{1}{2}\begin{pmatrix}\mathbf{x}_p-\boldsymbol{\mu}_p\\ \ln\left(\mathbf{T}_q-\mathbf{x}_q\right)-\boldsymbol{\mu}_q\end{pmatrix}^T\boldsymbol{\Sigma}^{-1}\begin{pmatrix}\mathbf{x}_p-\boldsymbol{\mu}_p\\ \ln\left(\mathbf{T}_q-\mathbf{x}_q\right)-\boldsymbol{\mu}_q\end{pmatrix}\right\}. \tag{4.82}$$

The descriptive statistics for this distribution are as follows; the mean is given by

$$\mathbf{x}_{mean,i}=\begin{cases}\mu_i, & i=1,2,\ldots,p,\\ T_i-\exp\left\{\mu_i+\dfrac{\sigma_i^2}{2}\right\}, & i=p+1,p+2,\ldots,N.\end{cases} \tag{4.83}$$

A multivariate median is given by;

$$\mathbf{x}_{median}=\begin{pmatrix}\boldsymbol{\mu}_p\\ \mathbf{T}_q-\exp\left\{\boldsymbol{\mu}_q\right\}\end{pmatrix}. \tag{4.84}$$

Finally we come to the mode;

$$\mathbf{x}_{mode}=\begin{pmatrix}\boldsymbol{\mu}_p-\langle\boldsymbol{\Sigma}_{pq},\mathbf{1}_q\rangle\\ \mathbf{T}_q-\exp\left\{\boldsymbol{\mu}_q-\langle\boldsymbol{\Sigma}_{qq},\mathbf{1}_q\rangle\right\}\end{pmatrix}. \tag{4.85}$$

We can see from (4.85) that this combination keeps the important property that we identified from the mixed Gaussian-lognormal distribution, that the mode associated with the Gaussian random variables becomes a function of the covariances between the Gaussian and reverse-lognormal random variables.

4.6.3 Combining With a Lognormal Distribution

As there exists the transform between the lognormal, the reverse lognormal, and the Gaussian distribution, it is straightforward to form any combination of these distributions to form a new mixed distribution. The motivation to consider a mixed lognormal-reverse-lognormal is due to, say moisture, not being the same distribution in a geographical area and if we are trying to minimize the errors in a data assimilation scheme, then we need the most consistent probability model to describe this.

Therefore, if we have p lognormally distributed random variables, where $x_i\in(0,\infty)$ for $i=1,2,\ldots,p$, and q reverse lognormally distributed random variables, such that $x_i\in(-\infty,T_i)$ for $i=p+1,p+2,\ldots,N$, then the associated mixed lognormal-reverse-lognormal PDF is given by

$$\begin{aligned}MXLRL\left(\mathbf{x}\right)=&\left(\prod_{i=1}^{p}\frac{1}{x_i}\right)\left(\prod_{i=p+1}^{N}\frac{1}{T_i-x_i}\right)\frac{1}{(2\pi)^{\frac{N}{2}}\left|\boldsymbol{\Sigma}\right|^{\frac{1}{2}}}\\ &\times\exp\left\{\frac{1}{2}\begin{pmatrix}\ln\mathbf{x}_p-\boldsymbol{\mu}_p\\ \ln\left(\mathbf{T}_q-\mathbf{x}_q\right)-\boldsymbol{\mu}_q\end{pmatrix}^T\boldsymbol{\Sigma}^{-1}\begin{pmatrix}\ln\mathbf{x}_p-\boldsymbol{\mu}_p\\ \ln\left(\mathbf{T}_q-\mathbf{x}_q\right)-\boldsymbol{\mu}_q\end{pmatrix}\right\}\end{aligned} \tag{4.86}$$

The descriptive statistics for this distribution are as follows; the mean is given by

$$\mathbf{x}_{mean,i} = \begin{cases} \exp\left\{\mu_i + \dfrac{\sigma_i^2}{2}\right\}, & i = 1, 2, \ldots, p, \\ T_i - \exp\left\{\mu_i + \dfrac{\sigma_i^2}{2}\right\}, & i = p+1, p+2, \ldots, N. \end{cases} \tag{4.87}$$

A multivariate median is given by;

$$\mathbf{x}_{median} = \begin{pmatrix} \exp\{\boldsymbol{\mu}_p\} \\ \mathbf{T}_q - \exp\{\boldsymbol{\mu}_q\} \end{pmatrix}. \tag{4.88}$$

Finally we come to the mode;

$$\mathbf{x}_{mode} = \begin{pmatrix} \exp\{\boldsymbol{\mu}_p - \langle \boldsymbol{\Sigma}, \mathbf{1}_N \rangle\} \\ \mathbf{T}_q - \exp\{\boldsymbol{\mu}_q - \langle \boldsymbol{\Sigma}, \mathbf{1} \rangle\} \end{pmatrix}. \tag{4.89}$$

An important feature to note here is that the mode induces covariances between all of the components of the mode, due to the fact that off set is the same for both the lognormal and reverse-lognormal modes, which comes from the products in front of the exponential.

4.6.4 Combining Multivariate Gaussian, Lognormal, and Reverse-Lognormal Distributions

We come to the final part for the multivariate reverse lognormal section, where we combine the PDFs of the multivariate reverse lognormal with those of a multivariate Gaussian distribution and a multivariate lognormal distribution. Therefore, we assume that we have a vector of random variables, where we have p that are Gaussian distributed, q that are lognormally distributed, and r that are reverse-lognormally distributed, such that $x_i \in (-\infty, \infty)$, for $i = 1, 2, \ldots, p$, $x_i \in (0, \infty)$ for $i = p+1, p+2, \ldots, p+q$ and $x_i \in (-\infty.T_i)$ for $i = p+q+1, p+q+2, \ldots, N$, where $N = p+q+r$. Thus the associated multivariate mixed Gaussian-lognormal-reverse-lognormal PDF is given by

$$\begin{aligned} MXGLRL(\mathbf{x}) = & \left(\prod_{i=p+1}^{p+q} \frac{1}{x_i}\right)\left(\prod_{i=p+q+1}^{N} \frac{1}{T_i - x_i}\right) \frac{1}{(2\pi)^{\frac{N}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \\ & \times \exp\left\{-\frac{1}{2}\begin{pmatrix} \mathbf{x}_p - \boldsymbol{\mu}_p \\ \ln \mathbf{x}_q - \boldsymbol{\mu}_q \\ \ln(\mathbf{T}_r - \mathbf{x}_r) - \boldsymbol{\mu}_r \end{pmatrix}^T \begin{pmatrix} \boldsymbol{\Sigma}_{pp} & \boldsymbol{\Sigma}_{pq} & \boldsymbol{\Sigma}_{pr} \\ \boldsymbol{\Sigma}_{qp} & \boldsymbol{\Sigma}_{qq} & \boldsymbol{\Sigma}_{qr} \\ \boldsymbol{\Sigma}_{rp} & \boldsymbol{\Sigma}_{rq} & \boldsymbol{\Sigma}_{rr} \end{pmatrix} \begin{pmatrix} \mathbf{x}_p - \boldsymbol{\mu}_p \\ \ln \mathbf{x}_q - \boldsymbol{\mu}_q \\ \ln(\mathbf{T}_r - \mathbf{x}_r) - \boldsymbol{\mu}_r \end{pmatrix}\right\} \end{aligned} \tag{4.90}$$

The descriptive statistics for this distribution are as follows; the mean is given by

$$\mathbf{x}_{mean,i} = \begin{cases} \mu_i, & i = 1, 2, \ldots, p \\ \exp\left\{\mu_i + \dfrac{\sigma_i^2}{2}\right\}, & i = p+1, 2, \ldots, p+q, \\ T_i - \exp\left\{\mu_i + \dfrac{\sigma_i^2}{2}\right\}, & i = p+q+1, p+q+2, \ldots, N. \end{cases} \tag{4.91}$$

A multivariate median is given by;

$$\mathbf{x}_{median} = \begin{pmatrix} \boldsymbol{\mu}_p \\ \exp\{\boldsymbol{\mu}_q\} \\ \mathbf{T}_r - \exp\{\boldsymbol{\mu}_r\} \end{pmatrix}. \tag{4.92}$$

Finally the mode is;

$$\mathbf{x}_{mode} = \begin{pmatrix} \boldsymbol{\mu}_p - \langle \boldsymbol{\Sigma}_{pq}, \mathbf{1}_q \rangle - \langle \boldsymbol{\Sigma}_{pr}, \mathbf{1}_r \rangle \\ \exp\{\boldsymbol{\mu}_q - \langle \boldsymbol{\Sigma}_{qq}, \mathbf{1}_q \rangle - \langle \boldsymbol{\Sigma}_{qr}, \mathbf{1}_r \rangle\} \\ \mathbf{T}_r - \exp\{\boldsymbol{\mu}_r - \langle \boldsymbol{\Sigma}_{rq}, \mathbf{1}_q \rangle - \langle \boldsymbol{\Sigma}_{rr}, \mathbf{1}_r \rangle\} \end{pmatrix}. \tag{4.93}$$

These additional mixed multivariate distributions since the first edition are still in their infancy, but have a lot of potential to better model moisture fields, as well as other positive definite variables that can change seasonally, monthly, or even dynamically. For some examples we recommend [142,158,159, 220,221] where there is evidence of distributions changing seasonally, but also dynamically between the three distributions we have combined here.

4.7 Gamma Distribution

Unlike with the multivariate Gaussian, lognormal, and mixed distributions, there is no single definition for a bivariate or multivariate Gamma distribution.

4.7.1 Bivariate Gamma Distribution

Here we shall present McKay's bivariate gamma distribution, which is defined by

$$p_{X_1,X_2}(x_1, x_2) = \frac{c^{a+b}}{\Gamma(a)\,\Gamma(b)} x_1^{a-1} (x_2 - x_1)^{a-1} e^{-cx_2}, \tag{4.94}$$

where $x_2 > x_1 > 0$ and the three parameters a, b, c are greater than zero. The marginal distribution for x_1 and $_2$ are univariate gamma distributions $\gamma_1(a)$ and $\gamma_2(a+b)$. The conditional probability density function is a beta distribution with parameters a and b. The correlation coefficient for McKay's bivariate distribution is given by $\sqrt{\dfrac{a}{a+b}}$.

The bivariate McKay gamma distribution has been used in hydrological applications in [65,66], where x_1 is assumed to represent annual stream flow while x_2 represents areal precipitation. The justification for the condition that $x_2 > x_1$ is based upon the physical argument that for water height basins with little over a year's storage, this approximation is true.

In hydrology, as well as in soil moisture modeling, we could have the following combinations of the random variables: $R = X_1 + X_2$, $K = X_1 X_2$ and $W = \dfrac{X_1}{X_1 + X_2}$, where if X_1 and X_2 denote the rainfall intensity and the duration of a storm, then the product of these two random variables would represent the amount of rainfall produced by that storm.

4.7.2 Multivariate Gamma Distribution

As with the bivariate gamma distribution, there is no one definition for a multivariate gamma distribution. We shall only present the generalized distribution. We assume that $X_0, X_1, \ldots, X_N$ are independent gamma-distributed random variables, where each random variable follows a univariate gamma distribution of the form

$$P_{x_i}(x_i) = \frac{1}{\Gamma(a_i)} e^{-x_i} x_i^{a_i - 1}, \tag{4.95}$$

where $x_i > 0$, $a_i > 0$, $i = 0, 1, \ldots, N$. Therefore, if we now form the new set of random variables as $Y_i = X_0 + X_i$ for $i = 1, 2, \ldots, N$, then the joint distribution of the new random variables is given by

$$P_{X_0, X_1, \ldots, X_N}(x_0, x_1, \ldots, x_N) = \frac{1}{\prod_{i=0}^{N} \Gamma(a_i)} e^{-\sum_{i=0}^{N} x_i} \prod_{i=0}^{N} x_i^{a_i - 1} \tag{4.96}$$

for $x_i > 0$, $a_i > 0$ and $i = 0, 1, \ldots, N$. Given the joint distribution in (4.96), we can obtain the joint density function of $(X_0, Y_1, \ldots, Y_N)$ as

$$p_{X_0, Y_1, \ldots, Y_N}(x_0, y_1, \ldots, y_n) = \frac{1}{\prod_{i=0}^{N} \Gamma(a_i)} x_0^{a_0 - 1} \left(\prod_{i=1}^{N} (y_i - x_0)^{a_i - 1} \right) e^{(N-1)x_0 - \sum_{i=1}^{N} y_i} \tag{4.97}$$

for $y_i \geq x_0 \geq 0$ for $i = 1, 2, \ldots, N$, which is now similar to the bivariate McKay gamma distribution. However, removing the x_0 requires the integration of a difficult term, and we recommend reading [224] for a full discussion on how this can be accomplished or avoided.

4.8 Summary

In this chapter, we have extended the definitions of the univariate Gaussian, lognormal, reverse-lognormal, and gamma PDFs to the bivariate and multivariate case. We have introduced a new type of distribution, the mixed Gaussian-lognormal distribution, and shown that the mode of this distribution links the Gaussian distributed random variables to the lognormal distributed random variables as

the part of the mode associated with the Gaussian random variables are a function of the covariances between the Gaussian and the lognormal random variables.

In this edition of the textbook we have introduced three new mixed multivariate distributions: the Gaussian-reverse-lognormal, the lognormal-reverse-lognormal, as well as the Gaussian-lognormal-reverse-lognormal distribution. We have presented the descriptive statistics for each of these distributions, and have shown that the mode retains the covariance structure between the different distributed random variables.

We have introduced the marginal and conditional distributions for some of the multivariate distributions considered in this chapter, where the multivariate gamma distribution may be difficult to express in this form; again we refer the readers to [224] for more detail about the multivariate gamma distributions. We have seen that for the non-univariate case, the minimum variance descriptive statistic, or the mean, has to be computed through its marginal distributions. We have shown for the non-gamma distributions presented here that the multivariate mode is found by differentiating the multivariate PDF through vector differentiation, and have shown that it is unique for these distributions.

In data assimilation we consider the multivariate formulation to obtain the cost function, or the equations for the moments, and as such we have introduced the multivariate distributions where there is a form of data assimilation associated with them, except the multivariate gamma, which we introduced to highlight the challenges ahead for data assimilation if the random variables we consider follow this type of distribution.

We now move on from the theory from probability, which is part of the basis of data assimilation, to introduce the theory for the calculus side of data assimilation.

This page intentionally left blank

CHAPTER

Introduction to Calculus of Variation 5

Contents

Calculus of variation is a very useful and powerful tool for solving differential equations, maximum surface area problems, and the largest area enclosed by a perimeter of a specific length, to name but a few examples. As the name suggests, calculus of variation plays an important part in the derivation of variational-based data assimilation systems. In this chapter we shall introduce the tools that will be used to derive the three- and four-dimensional variational data assimilation systems later.

5.1 Examples of Calculus of Variation Problems

5.1.1 Shortest/Minimum Distance

A typical example of a calculus of variations problem is: shortest distance problems. i.e. what is the shortest distance between points A and B? Fig. 5.1 shows an illustration of this type of problem. Given the locations of points A and B, what curve minimizes the distance between them?

To solve this problem, we first formulate a general problem for the distance from A to B for any curve y, and then seek to minimize this distance. As we can see in Fig. 5.2 the change in distance S, δS, is approximately related to the change in x, δx, and the change in y, δy, through a right-angle triangle. Therefore, by Pythagoras's theorem we can relate the change in distance squared to the square of the

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00009-7

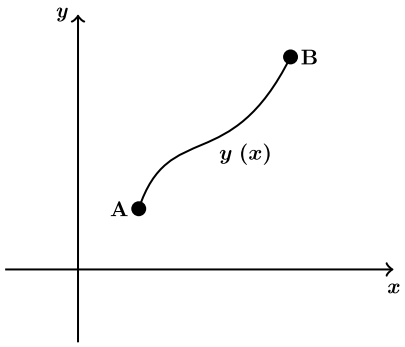


FIGURE 5.1

Illustration of a shortest distance calculus of a variation problem.

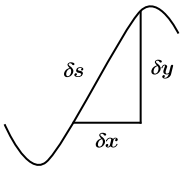


FIGURE 5.2

Illustration of change in distance for a shortest distance calculus of a variation problem.

change in x and the square of the change in y by

$$\delta S^2 \approx \delta x^2 + \delta y^2. \tag{5.1}$$

Dividing (5.1) on both sides by δx^2 results in

$$\left(\frac{\delta S}{\delta x}\right)^2 \approx 1 + \left(\frac{\delta y}{\delta x}\right)^2. \tag{5.2}$$

Taking the limit as $\delta x \to 0$ makes the expression in (5.2) equivalent, and it can now be written in terms of differential operators as

$$\left(\frac{dS}{dx}\right)^2 = 1 + \left(\frac{dy}{dx}\right)^2. \tag{5.3}$$

The total length from A to B is given, integrating (5.3) from a and b as

$$S[y] = \int_a^b dS \equiv \int_a^b \sqrt{1 + \left(\frac{dy}{dx}\right)^2}\, dx. \tag{5.4}$$

Therefore, for the differential equation in (5.4) the problem is to determine the specific $y(x)$ such that (5.4) is the least over all possible y, given the boundary conditions $y(a) = \alpha$ and $y(b) = \beta$, and where $A(a, \alpha)$ and $B(b, \beta)$ are given. These types of problems are referred to as **geodesics**.

5.1.2 Brachistochrone Problem

The next classical example is the **Brachistochrone Problem**. As shown in Fig. 5.3, the problem is to find the shape of a smooth wire passing from point 0 to B such that the time taken for a bead to slide down is the quickest/least. This is one of the oldest calculus of variation problems that is referred to as such, and can be traced back to Sir Isaac Newton in 1696.

The starting point is to consider the equation for speed, v, where $v = \frac{dS}{dt}$, with t denoting time. This can easily be rearranged so that $dt = \frac{dS}{v}$. This then leads to the total time being

$$T[y] = \int_0^b \frac{dS}{v}. \tag{5.5}$$

We know from the first example that $dS = \sqrt{1 + y'^2}\,dx$, where $y' \equiv \frac{dy}{dx}$. The next step is to find a way to express the velocity, v, in terms of y. This is achieved through the application of the conservation of energy, specifically the equation that describes the change of kinetic energy to gravitational potential energy,

$$\frac{1}{2}mv^2 = mgy \Rightarrow v = \sqrt{2gy}, \tag{5.6}$$

where m is mass and g is the acceleration due to the gravity constant. This enables us to write the functional for total time in terms of y as

$$T[y] = \frac{1}{\sqrt{2g}} \int_a^b \frac{\sqrt{1 + y'^2}}{\sqrt{y}}\,dx, \qquad \begin{matrix} y(0) = 0 \\ y(b) = \beta \end{matrix}. \tag{5.7}$$

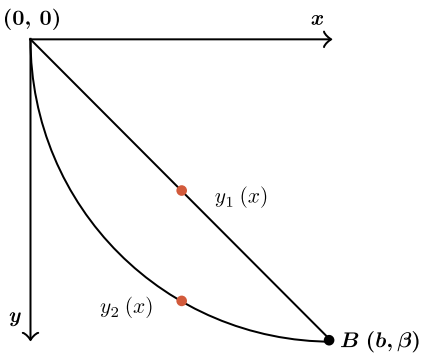


FIGURE 5.3

Illustration of the path with the quickest time to travel, referred to as the Brachistochrone Problem.

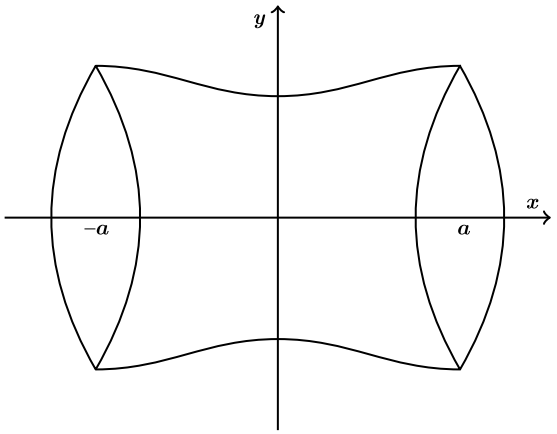


FIGURE 5.4

Illustration of a maximum surface area problem.

5.1.3 Minimum Surface Area

The next class of calculus of variation problems are those related to minimizing **surface area**. If we consider the diagram in Fig. 5.4, then we can see that we are seeking the curve, $y(x)$, that fits between the two boundary point values so that when the curve is rotated about the x-axis 360 degrees, we obtain a curved surface. Therefore, we need to find the curve such that the resulting surface area is minimized.

The mathematical starting point is the surface area equation;

$$SA[y] \approx \sum 2\pi y \delta S. \tag{5.8}$$

Taking the limit as $\delta s \to 0$ implies

$$SA[y] = 2\pi \int_{-a}^{a} y dS = 2\pi \int_{-a}^{a} y\sqrt{1+y'^2}dx. \tag{5.9}$$

Therefore, we seek the function y that minimizes the functional in (5.9). This problem is related to what is often called the **soap bubble problem**.

5.1.4 Dido's Problem—Maximum Enclosed Area for a Given Perimeter Length

The fourth class of problems that we consider are related to maximum area enclosed by a curve of a given length from points $A = (a, \alpha)$ to point $B = (b, \beta)$. A classical version of this type of problem is referred to as **Dido's problem**. The problem is illustrated in Fig. 5.5. The functional for area A that has to be maximized is given by

$$A[y] = \int_{a}^{b} y dx, \tag{5.10}$$

given $y(a) = y(b) = 0$, subject to the distance constraint $\int_{a}^{b} \sqrt{1+y'^2}dx$.

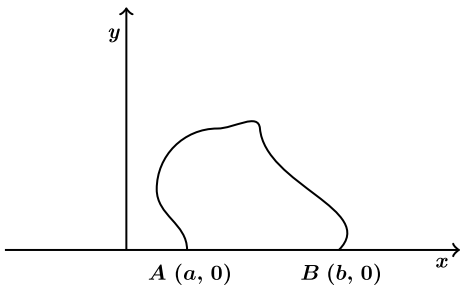


FIGURE 5.5

Illustration of the Dido maximum area enclosed problem.

5.1.5 General Form of Calculus of Variation Problems

Given the four examples above, we have seen that the problems can be a function of x, y, and y', so we can express the general form of a one-dimensional, single variable calculus of variation problem as

$$I[y] = \int_a^b F\left(x, y, y'\right) dx, \qquad \begin{matrix} y(a) = \alpha \\ y(b) = \beta \end{matrix} . \tag{5.11}$$

The problem defined in (5.11) is referred to as the **functional**. It is a "functional form" mapping between sets of functions to numbers, where an example would be

$$I[y] = \int_a^b \sqrt{1 + y'^2} dx, \quad I[3x] = \int_0^2 \sqrt{1+9} dx = \sqrt{40}.$$

Therefore, $I : 3x \to \sqrt{40}$. The standard calculus of variation problem is to find the particular y, denoted y_0 for which $I[y]$ is least/greatest overall possible and suitable smooth functions, satisfying $y(a) = \alpha$ and $y(b) = \beta$.

5.2 Solving Calculus of Variation Problems

The starting point to solve (5.11) is to assume that the function y is such that $y \in \mathbb{C}^2[a, b]$, where $\mathbb{C}$ represent the complex number plane, the superscript 2 refers to the function being continuously differentiable to its second derivative and the $[\cdot]$ refers to the range that the function is bounded by and obtains those boundaries.

It is now assumed that there is a local minimum when $y = y_0(x)$. This minimum is referred to as the **extremum**. It should also be noted that $y_0(x)$ may not be unique. In data assimilation it is assumed that the solution is at the global minimum, but $y_0(x)$ is a local minimum if $I[y] \geq I[y_0]$ for all $y(x)$ sufficiently close to $y_0(x)$, where $y \in \mathbb{C}^2[a, b]$ and $y(a) = \alpha$ and $y(b) = \beta$.

The next step is to assume a **variation**, which is small perturbation to the function, $y(x)$, of the form $y(x) = y_0 + \varepsilon\eta(x)$, where $\eta(x)$ is referred to as an **admissible variation**. We require $\eta(x)$ to have the following properties: (1) $\eta(x) \in \mathbb{C}^2[a, b]$, (2) $\eta(a) = 0$ and $\eta(b) = 0$. The second property

arises because we have $y(a) = \alpha = y_0(a) + \varepsilon\eta(a) = \alpha + \varepsilon\eta(a)$, which can only hold if $\eta(a) = 0$. The same is also true at the b point. A simple illustration of this condition is presented in Fig. 5.6.

Therefore, we require $I[y_0 + \varepsilon\eta(x)] \geq I[y_0]$ if y_0 is an extremum, for all admissible variations η and for sufficiently small ε. Another way to consider this local minimum condition is

$$\int_a^b F\left(x, y_0 + \varepsilon\eta(x), y_0' + \varepsilon\eta'(x)\right) dx - \int_a^b F\left(x, y_0, y_0'\right) dx \to 0, \tag{5.12}$$

for all sufficiently small ε.

The next step is to recall the Taylor series expansion formula, but for $f(x_0 + h) - f(x_0)$, which is

$$\begin{aligned} f(x_0 + h) - f(x_0) &= f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + \cdots - f(x_0) \geq 0, \\ &= hf'(x_0) + \frac{h^2}{2} f''(x_0) + \cdots + \geq 0, \\ &\Rightarrow f'(x_0) = 0. \end{aligned}$$

The Taylor series expansion can be applied to functions of more than one variable. Although the problems that have been introduced so far are for one variable. The Taylor series expansion for a function of two variables is given by

$$G(x + h, y + k) = G(x, y) + hG_x(x, y) + hG_y(x, y) + \text{second-order terms}. \tag{5.13}$$

The next step is to apply (5.13) to $F\left(x, y_0(x) + \varepsilon\eta(x), y_0'(x) + \varepsilon\eta'(x)\right)$, and then substitute into (5.12). The result of this substitution is

$$F\left(x, y_0(x) + \varepsilon\eta(x), y_0'(x) + \varepsilon\eta'(x)\right) \approx F\left(x, y_0, y_0'\right) + \varepsilon\eta(x) F_y\left(x, y_0, y_0'\right) + \varepsilon\eta'(x) F_{y'}\left(x, y_0, y_0'\right) + O\left(\varepsilon^2\right). \tag{5.14}$$

This implies that $I[y_0 + \varepsilon\eta(x)] - I[y_0]$ is

$$I[y_0 + \varepsilon\eta(x)] - I[y_0] = \int_a^b \varepsilon\eta(x) F_y\left(x, y_0, y_0'\right) + \varepsilon\eta'(x) F_{y'}\left(x, y_0, y_0'\right) dx + O\left(\varepsilon^2\right) \geq 0. \tag{5.15}$$

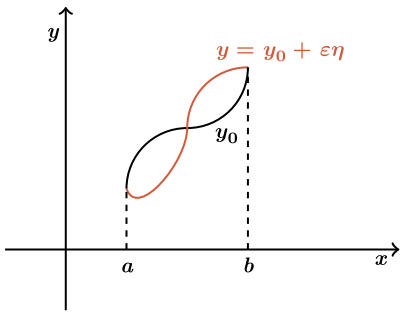


FIGURE 5.6

Illustration of an admissible variation.

Another way of expressing (5.15) is as the change of the functional, ΔI, as

$$\Delta I = I[y_0 + \varepsilon\eta] - I[y_0], \quad \Delta I = \varepsilon\delta I + O(\varepsilon^2), \quad \delta I = \int_a^b \eta F_y + \eta' F_{y'} dx. \tag{5.16}$$

With the assumption that y_0 is an extremum, we require that ΔI should be one signed for all variations, η, and for sufficiently small ε. However, for sufficiently small ε, the sign of ΔI is dictated by the sign of $\varepsilon\delta I$, and we can choose a small enough ε such that the $O(\varepsilon^2)$ terms can be made smaller than $\varepsilon\delta I$. The next condition is that unless $\delta I = 0$, we can arbitrarily choose different values for ε such that the sign of δI can be changed. This leads to the condition that

$$\int_a^b \eta F_y + \eta' F_{y'} dx = 0, \quad \forall \eta \in \mathbb{C}^2[a,b]. \tag{5.17}$$

This then leads to a very important lemma associated with calculus of variation, referred to as **Lagrange's Lemma**.

Lemma 5.1. ***Lagrange's Lemma****: If* $f \in \mathbb{C}^2[a,b]$ *and* $\int_a^b \eta f dx = 0$*, then* $\forall \eta \in \mathbb{C}^2[a,b]$ *such that* $\eta(a) = \eta(b) = 0$ *then* $f \equiv 0$.

Proof. Assume $f \geq 0$ somewhere such that $f(x) > 0$ in $\left[\hat{a}, \hat{b}\right]$ by continuity. Let

$$\eta = \begin{cases} \left(\hat{b} - x\right)^3 (x - \hat{a})^3 & x \in \left(\hat{a}, \hat{b}\right) \\ 0 & x \notin \left(\hat{a}, \hat{b}\right) \end{cases} \quad \eta(a) = \eta(b) = 0, \quad \eta \in \mathbb{C}^2[a,b].$$

Therefore

$$\int_a^b \eta f dx = \int_{\hat{a}}^{\hat{b}} \left(\hat{b} - x\right)^3 (x - \hat{a})^3 f(x) dx > 0.$$

Returning to $\delta I = 0$, we have

$$\int_a^b \eta F_y + \eta' F_{y'} dx = 0. \tag{5.18}$$

Applying integration by parts to the second term in (5.18) leads to

$$\int_a^b \eta F_y dx + \eta F_{y'}\Big|_a^b - \int_a^b \eta \frac{d}{dx} F_{y'} dx = 0. \tag{5.19}$$

An important feature to note here is that the second term in (5.19) is equal to zero due to the condition that all admissible variations satisfy the boundary conditions, $\eta(a) = \eta(b) = 0$. Factorizing η from the remaining terms in (5.19) leads to

$$\int_a^b \eta \left(F_y - \frac{d}{dx} F_{y'}\right) dx = 0. \tag{5.20}$$

By Lagrange's Lemma, we have that

$$F_y - \frac{d}{dx} F_{y'} = 0, \tag{5.21}$$

because the equation in (5.20) has to be zero for all admissible variations η, for the functional $I[y]$, not just $\eta = 0$. The equation in (5.21) is referred to as **Euler's equation**, which is a second-order ordinary differential equation for y_0. The next step is to recall that F is evaluated at y_0; this means above (5.21) can be written as

$$F_y\left(x, y_0(x), y_0'(x)\right) - \frac{d}{dx} F_{y'}\left(x, y_0, y_0'\right) = 0, \tag{5.22}$$

with $y_0(a) = \alpha$, $y_0(b) = \beta$, where $F_y \equiv \frac{\partial F}{\partial y}$ and $F_{y'} \equiv \frac{\partial F}{\partial y'}$. This leads to the following theorem:

Theorem 5.2. *If* $I[y] = \int_a^b F\left(x, y, y'\right) dx$ *has an extremum* $y_0(x)$, *where* $y(x) \in \mathbb{C}^2[a,b]$, $y(a) = \alpha$, $y(b) = \beta$ *and* F *has continuous partial derivatives up to and including the second derivative, then*

$$\frac{\partial F}{\partial y} - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial y'}\right) = 0,$$

which is Euler's equation at $y_0(x)$.

Proof. Let $y = y_0 + \varepsilon\eta$ be an admissible variation so that $\eta(a) = \eta(b) = 0$, where $\eta \in \mathbb{C}^2[a,b]$. The change in the functional, ΔI, is

$$\begin{aligned}
\Delta I &= I[y_0 + \varepsilon\eta] - I[y_0], \\
&= \int_a^b F\left(x, y_0 + \varepsilon\eta, y_0' + \varepsilon\eta'\right) dx - \int_a^b F\left(x, y_0, y_0'\right) dx, \\
&= \int_a^b F + \varepsilon\eta F_y + \varepsilon\eta' F_{y'} dx + O\left(\varepsilon^2\right) - \int_a^b F\, dx, \\
&= \varepsilon\delta I + O\left(\varepsilon^2\right).
\end{aligned}$$

As we saw earlier, for a local extremum, ΔI should be one signed for all admissible η and sufficiently small ε, thus $\delta I = 0$, which implies

$$\begin{aligned}
\delta I &\equiv \int_a^b \eta F_y + \eta' F_{y'} dx = 0, \\
&= \int_a^b \eta\left(F_y - \frac{d}{dx} F_{y'}\right) dx + \eta F_{y'}\Big|_a^b = 0, \\
&\Rightarrow F_y - \frac{d}{dx} F_{y'} = 0,
\end{aligned}$$

by Euler's equation.

Note: This is only a necessary condition. The solutions to Euler equations are referred to as **extremals**. It should also be noted that if a function is an extremum then it is also an extremal, but the

opposite is not always the case. This then indicates that all solutions to Euler equations need to be checked to see if they are extremums.

We now return to the example from Section 5.1.1. We recall that the functional to be minimized is

$$I[y] = \int_0^1 1 + y'^2\, dx \qquad \begin{matrix} y(0) = 0 \\ y(1) = 1 \end{matrix},$$

which implies that we are seeking an extremal in $\mathbb{C}^2[0, 1]$.

The first step is to form the Euler equation for (5.4):

$$\begin{aligned} & F \equiv 1 + y'^2, \quad F_y = 0, \quad F_{y_0'} = 2y'. \\ & EE : 0 - \frac{d}{dx} 2y' = 0, \\ & y_0'(x) = A \Rightarrow y_0(x) = B + Ax, \\ & \begin{matrix} y_0(0) = 0 \\ y_0(1) = 1 \end{matrix} \Rightarrow \begin{matrix} B = 0 \\ A = 1 \end{matrix} \Rightarrow y_0(x) = x. \end{aligned}$$

Therefore, we have that $y_0 = x$ is an extremal, but is it an extremum? To ascertain if $y_0 = x$ is an extremum, we consider $\Delta I = I[y_0 + \varepsilon\eta] - I[y_0]$ for $\eta \in \mathbb{C}^2[0, 1]$ and $\eta(0) = \eta(1) = 0$, which substituting $y_0 = x$ results in

$$\begin{aligned} \Delta I &= I[x + \varepsilon\eta] - I[x], \\ &= \int_0^1 \left(1 + (1 + \varepsilon\eta')^2\right) dx - \int_0^1 1 + 1^2\, dx, \\ &= \int_0^1 1 + 1 + 2\varepsilon\eta' + \varepsilon^2\eta'^2\, dx - 2, \\ \Rightarrow \Delta I &= 2\varepsilon \underbrace{\int_0^1 \eta'\, dx}_{\delta I = 0} + \varepsilon^2 \int_0^1 \eta'^2\, dx, \\ \Delta I &= \varepsilon^2 \int_0^1 \eta'^2\, dx > 0 \quad \forall \eta. \end{aligned} \tag{5.23}$$

It should also be noted, the reason for the greater than zero sign is because the integral is square and therefore is always positive for any η, as well as for sufficiently small ε, but it is for all ε, as ΔI is a function of ε^2, which implies that this solution is a global minimum. The value of the functional at $y_0 = x$ is $I[x] = 2$. Therefore, the curve that gives the shortest distance between points A and B is a straight line, $y_0 = x$.

Example 5.3. *Find the minimum of* $I[y] = \int_0^{\frac{\pi}{2}} \left(y'^2 - y^2 + 2xy\right) dx$ *with* $y(0) = y\left(\frac{\pi}{2}\right) = 0$.

The starting point is to recall Euler's equation $F_{y_0} - \frac{d}{dx} F_{y_0'} = 0$, which implies

$$F_{y_0} = -2y_0 + 2x, \quad F_{y_0'} = 2y_0'.$$

Substituting the two terms above into Euler's equation results in

$$-2y_0 + 2x - \frac{d}{dx}\left(2y_0'\right) = 0.$$

Applying the differential operator results in the following second-order ordinary differential equation to solve for y_0:

$$y_o'' + y_0 = x. \tag{5.24}$$

Recalling solving techniques for these types of ordinary differential equation problems, then we know that we require a complementary function (CF) and a particular integral (PI), which can easily be shown to be

$$y_o = \underbrace{A\cos(x) + B\sin(x)}_{\text{CF}} + \underbrace{x}_{\text{PI}}. \tag{5.25}$$

Evaluating the boundary conditions $y_0(0) = 0$ and $y_0\left(\frac{\pi}{2}\right) = 0$ gives us $A = 0$ and $B = -\frac{\pi}{2}$. Therefore, the extremal for the functional in Example 5.3 is

$$y_0(x) = x - \frac{\pi}{2}\sin(x). \tag{5.26}$$

Exercise 5.4. *Prove whether or not* (5.26) *is an extremum for the functional in Example 5.3.*

Exercise 5.5. *Find the extremals, if any, of*

- $I[y] = \int_1^2 x^2 y'^2\, dx,\ y(1) = 1,\ y(2) = \frac{1}{2},$ *in* $\mathbb{C}^2[1,2]$*; and*
- $I[y] = \int_1^2 x^2 y'^2\, dx,\ y(-1) = 1,\ y(2) = 1,$ *in* $\mathbb{C}^2[-1,1]$.

Exercise 5.6. *has an extremal* $y_0 = \frac{3x}{4}$. *Deduce that* y_0 *is an extremum by showing that it is a local minimum Show that the functional* $I[y] = \int_0^1 \left(y'^2 - 1\right)^2 dx,\ y(0) = 0,\ y(1) = \frac{3}{4},\ y \geq 0$ *where* $y \in \mathbb{C}^2[0,1]$

Exercise 5.7. *Find the general form for the extremals of the following functionals.*

1. $I[y] = \int_a^b y^2 + y'^2 - 2y\sin x\, dx$, *where* $y(a)$, $y(b)$ *are given.*
2. $I[y] = \int_a^b y^2 + y'^2 - 2ye^x\, dx$, *where* $y(a)$, $y(b)$ *are given.*

5.2.1 Special Cases for Euler's Equations

In this subsection we consider three cases when certain terms are missing from the functional, that will have an impact on the type of differential equation that has to be solved from the associated Euler Equation.

- **Case One:** $F\left(x, y, y'\right) = F\left(x, y\right)$. This is the case where y' is missing, and so Euler equation becomes

$$F_y - \frac{d}{dx}0 = 0 \Rightarrow F_y = 0. \tag{5.27}$$

- **Case Two:** $F\left(x, y, y'\right) = F\left(x, y'\right)$. This is the case where y is missing, and so the associated Euler equation here is

$$0 - \frac{d}{dx}F_{y'} = 0 \Rightarrow F_{y'} = C. \tag{5.28}$$

- **Case Three:** $F\left(x, y, y'\right) = F\left(y, y'\right)$. This is the case where x is missing. However, it is not straightforward what changes occur to Euler's equation. The starting point is to consider the chain rule applied to $\frac{dF}{dx}$ in terms of y and y'. Therefore

$$\begin{aligned}\frac{dF}{dx} &= \frac{dy}{dx}\frac{\partial F}{\partial y} + \frac{dy'}{dx}\frac{\partial F}{\partial y'},\\ &= y'F_y + y''F_{y'}.\end{aligned}$$

The next step is to rearrange Euler's equation so that $F_y = \frac{d}{dx}F_{y'}$. This leads to

$$\begin{aligned}\frac{dF}{dx} &= y'\frac{d}{dx}F_{y'} + y''F_{y'},\\ &= \frac{d}{dx}\left(y'F_{y'}\right),\\ &\Rightarrow \frac{d}{dx}\left(F - y'F_{y'}\right) = 0,\\ &\Rightarrow F - y'F_{y'} = C, \quad \forall x.\end{aligned} \tag{5.29}$$

The differential equation in (5.29) is referred to as the **first integral**.

We shall now consider two examples, where the associated functions are of the third type above.

Solving the Brachistochrone Problem

The first example is the Brachistochrone Problem. This problem is to find the minimum of

$$I\left[y\right] = \frac{1}{\sqrt{2g}}\int_0^b \frac{\sqrt{1+y'^2}}{\sqrt{y}}dx. \tag{5.30}$$

The first feature to notice is that (5.30) is not dependent on x. This implies that we have to solve the first integral of the Euler equation, which is derived as follows:

$$\frac{\sqrt{1+y'^2}}{\sqrt{y}} - y'\frac{\partial}{\partial y'}\left(\frac{\sqrt{1+y'^2}}{\sqrt{y}}\right) = C,$$

$$\frac{\sqrt{1+y'^2}}{\sqrt{y}} - y'\frac{1}{\sqrt{y}}\frac{1}{2}\frac{1}{\sqrt{1+y'^2}}2y' = C,$$

$$\frac{\sqrt{1+y'^2}}{\sqrt{y}} - \frac{y'^2}{\sqrt{y}\sqrt{1+y'^2}} = C,$$

$$1 + y'^2 - y'^2 = C\sqrt{y}\sqrt{1+y'^2},$$

$$1 = C\sqrt{y}\sqrt{1+y'^2}. \tag{5.31}$$

The next step is to square (5.31) and rearrange to isolate y', which results in

$$y' = \pm\sqrt{\frac{1}{C^2 y} - 1},$$

$$\frac{dy}{dx} = \pm\sqrt{\frac{1}{C^2 y} - 1} = \frac{\sqrt{1-C^2 y}}{C\sqrt{y}},$$

$$\Rightarrow \int \frac{C\sqrt{y}}{\sqrt{1-C^2 y}}dy = \int 1dx + A,$$

$$\int \frac{C\sqrt{y}}{\sqrt{1-C^2 y}}dy = x + A. \tag{5.32}$$

Next we introduce the change of variable, $C^2 y = \sin^2\theta$. This implies that $y = \frac{\sin^2\theta}{C^2}$. The next step is to derive the change of dy into $d\theta$, which can be shown to be $dy = \frac{2\sin\theta\cos\theta}{C^2}d\theta$. The change of variable, and the equivalent change to the integrand, now are substituted in (5.32), which results in

$$\int \frac{C\frac{\sin\theta}{C}}{\sqrt{1-\sin^2\theta}}\frac{2\sin\theta\cos\theta}{C^2}d\theta = x + A,$$

$$\frac{1}{C^2}\int 2\sin^2\theta d\theta = x + A,$$

$$\frac{1}{C^2}\int (1-\cos 2\theta)\, d\theta = x + A,$$

$$\Rightarrow \frac{1}{2C^2}(2\theta - \sin 2\theta) = x + A.$$

Therefore, we have equations for y and x that are both functions of θ, which means that the extremal has a parametric form

$$x = x(\theta) = \frac{1}{2C^2}(2\theta - \sin 2\theta) - A, \tag{5.33}$$

$$y = y(\theta) = \frac{\sin^2\theta}{C^2}. \tag{5.34}$$

The next step is to solve for the constants A and C. We start by substituting the first boundary condition, $(x, y) = (0, 0)$ into (5.34), which results in

$$0 = \frac{\sin^2 \theta_0}{C^2}.$$

There are several angles θ that will make $\sin^2 \theta = 0$, so without loss of generality, we shall pick $\theta_0 = 0$. Given this choice for θ_0, we now substitute this value into (5.33), which results in $A = 0$.

The next step is to rewrite (5.33) in terms of $\cos 2\theta$ through the trigonometric identity, $\sin^2 \theta \equiv \frac{1-\cos 2\theta}{2}$. This then makes the parametric equations

$$x = x(\theta) = \frac{1}{2C^2}(2\theta - \sin 2\theta), \tag{5.35}$$

$$y = y(\theta) = \frac{1}{2C^2}(1 - \cos 2\theta). \tag{5.36}$$

The next step is to use the second boundary condition $(x, y) = (b, \beta)$, when $\theta = \theta_1$. To solve for θ_1, we take the ratio of (5.35) to (5.36) evaluated at $x = b$ and $y = \beta$, respectively, which results in

$$\frac{2\theta_1 - \sin 2\theta_1}{1 - \cos 2\theta_1} = \frac{b}{\beta}. \tag{5.37}$$

We then have to ask the question: what curve is described by the parametric form

$$x(\theta) = \frac{1}{2C^2}(2\theta - \sin 2\theta),$$
$$y(\theta) = \frac{1}{2C^2}(1 - \cos 2\theta).$$

The answer is a cycloid, which is of the form

$$x(\psi) = R(\psi - \sin \psi),$$
$$y(\psi) = R(1 - \cos \psi),$$

where $R = \frac{1}{2C^2}$ is the radius of the rolling circle and $\psi = 2\theta$ is the angle turned from the origin to the point b. An illustration of the solution is shown in Fig. 5.7.

Therefore, the fastest path from the origin to the point (b, β) is not a straight line, as in the plane example, but is the path traced by a circle rolling along the x-axis. So on the distance along the curve,

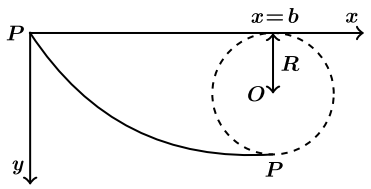


FIGURE 5.7

Illustration of the solution to the Brachistochrone Problem.

the acceleration due to gravity acting down a steeper initial slope enables the bead to travel faster than on a direct constant gradient to the point (b, β).

Solving the minimum surface area

The curve $y = y(x)$, $x \in [-a, a]$, with boundary conditions $y(a) = y(-a) = \alpha$, for $\alpha > 0$ is rotated about the x-axis. The surface area of the resulting shape is

$$I[y] = \int_{-a}^{a} 2\pi \sqrt{1 + y'^2}\, dx.$$

Therefore, we have to find the curve that satisfies the boundary conditions, so that when it is rotated about the x-axis, the surface area is minimized.

The first feature to notice is that the functional is independent of x, and as such we have to use the first integral approach again. Therefore, we need to form

$$F - y' F_{y'} = C' \Rightarrow 2\pi \sqrt{1 + y'^2} - y' \frac{2\pi y y'}{\sqrt{1 + y'^2}} = C'.$$

Multiplying throughout by $\sqrt{1 + y'^2}$ and defining $C = \frac{C'}{2\pi}$, we have

$$y\left(1 + y'^2\right) - y y' = C\sqrt{1 + y'^2} \Rightarrow y = C\sqrt{1 + y'^2}. \tag{5.38}$$

Squaring both sides of the second equation above, rearranging to isolate y', and then taking the square root, results in

$$\frac{dy}{dx} = \pm\sqrt{\frac{y^2}{C^2} - 1} \Rightarrow \pm \int \frac{dy}{\sqrt{\frac{y^2}{C^2} - 1}} = \int dx + A. \tag{5.39}$$

We now introduce a change of variable. Let $y = \cosh\theta$, so that $dy = C \sinh\theta\, d\theta$; this then makes $\frac{y^2}{C^2} - 1 = \cosh^2\theta - 1 = \sinh^2\theta$. Therefore,

$$\int \frac{dy}{\sqrt{\frac{y^2}{C^2} - 1}} \equiv \int \frac{C \sinh\theta d\theta}{\sinh\theta} = C\theta \equiv C \cosh^{-1}\left(\frac{y}{c}\right).$$

Evaluating the right-hand side integral in (5.39) and inverting the inverse of cosh results in an expression for y as

$$y = C \cosh\left(\pm\frac{(x + A)}{C}\right).$$

Using the property of the hyperbolic trigonometric functions that $\cosh -\theta = \cosh\theta$ implies that the general solution to the minimum surface area problem is

$$y = C \cosh\left(\frac{(x + A)}{C}\right). \tag{5.40}$$

The next step is to find values for the constants A and C. Applying the boundary conditions $y(a) = y(-a) = \alpha$ implies that

$$\alpha = Cy = C\cosh\left(\frac{(\alpha + A)}{C}\right) = C\cosh\left(\frac{(-\alpha + A)}{C}\right),$$
$$C\left(\cosh\left(\frac{(\alpha + A)}{C}\right) - \cosh\left(\frac{(A - \alpha)}{C}\right)\right) = 0. \tag{5.41}$$

For (5.41), we cannot have $C = 0$, as it would imply that $y \equiv 0$ from (5.38), which would not satisfy the boundary conditions, therefore $C \neq 0$. To carry on, we shall use the identity

$$\cosh p - \cosh q \equiv 2\sinh\frac{1}{2}(p + q)\sinh\frac{1}{2}(p - q),$$

which implies that (5.41) can be rewritten as

$$2\sinh\frac{A}{C}\sinh\frac{\alpha}{C} = 0. \tag{5.42}$$

As $\alpha \neq 0$, this implies that $\sinh\frac{A}{C} = 0$, which occurs for $\frac{A}{C} = 0 \Rightarrow A = 0$. Therefore, the extremal is

$$y_0(x) = C\cosh\frac{x}{c}, \tag{5.43}$$

where C is determined by α and a. Evaluating (5.43) at the point (a, α) implies that we have the situation $\alpha = C\cosh\frac{a}{C}$ which may or may not have roots for C, depending on a and α. Introducing the constant $D = \frac{a}{C}$, where D is to be determined, enables us to write $\cosh D = \frac{a}{C} = \frac{\alpha}{a}D$. Thus we need to find a root of $\cosh x = kx$, where $k = \frac{\alpha}{a}$.

We have plotted $\cosh x$ against three different values for k that are less than, equal to, and greater than k_{crit} where we shall prove the value of k_{crit} in Fig. 5.8. We can see from this figure that for $k < k_{crit}$, the line kx does not cross $\cosh x$, which implies that there are no extremals. When $k = k_{crit}$ there is one extremal, which can be shown to be an extremum. For $k > k_{crit}$ there are two extremals, where one of them is an extremum.

Now to determine the value for k_{crit}: when $k = k_{crit}$, $y = kx$ touches $y = \cosh x$. At point x_0 we have that the slope of $y = kx$ and $y = \cosh x$ are the same. Since $\frac{d}{dx}\cosh x = \sinh x$ and $\frac{d}{dx}kx = k$. Therefore, $k_{crit} = \sinh x_0$. We consider the distance from the origin to x_0 as OL and the distance LP as the distance from x_0 to $k_{crit}x = \cosh x$. Thus, forming the ratio $\frac{LP}{OL}$, we have

$$\frac{LP}{OL} \Rightarrow \frac{k_{crit}x_0}{x_0} = \frac{\cosh x_0}{x_0} \Rightarrow k_{crit} = \frac{\cosh x_0}{x_0}. \tag{5.44}$$

From earlier, we also have that $k_{crit} = \sinh x_0$, thus equating this expression with (5.44) and rearranging gives an expression for x_0 as $x_0 = \coth x_0$, where $\coth x$ is the hyperbolic cotangent function and only has one root at $x_0 = 1.19967\ldots$. Substituting this value for x_0 into the equation for $k_{crit} = \sinh x_0 \Rightarrow k_{crit} = 1.50886\ldots$.

In summary, if

- $\frac{\alpha}{a} > 1.50886\ldots$ then there are two extremals, only one is a extremum that corresponds to the smaller root of $kx = \cosh x$;

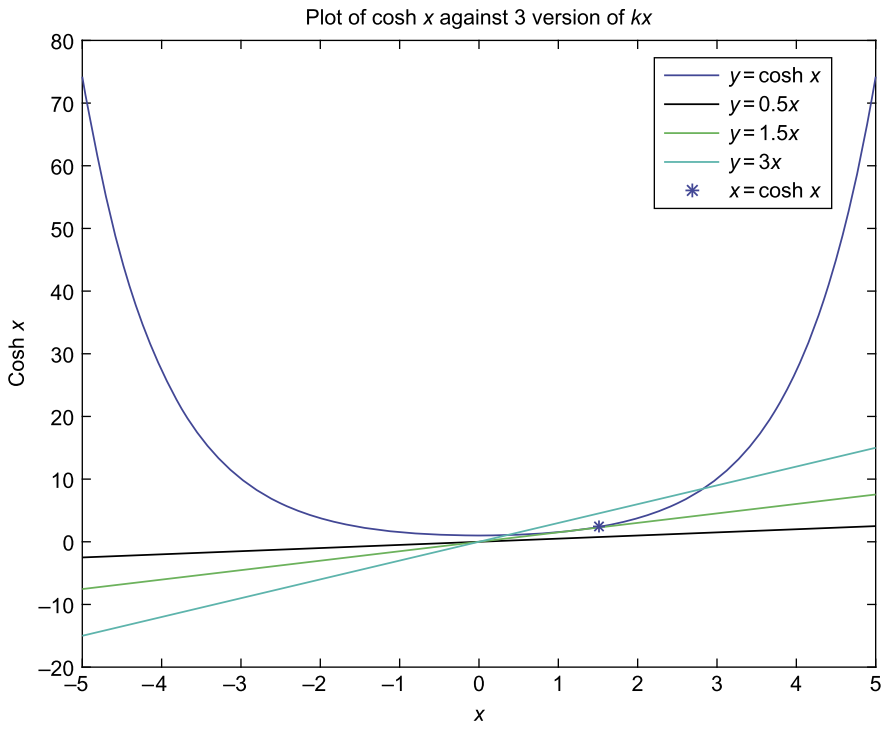


FIGURE 5.8

Plot of $\cosh x$ with three different values for kx to illustrate how many extremals there are when k is either side of, or equal to, the critical value.

- $\frac{\alpha}{a} = 1.50996\ldots$ then there is one extremal which is also an extremum; and
- $\frac{\alpha}{a} < 1.50886\ldots$ then there are no extremals.

The extremals of this problem are referred to as catenaries and the curve that an idealized hanging chain or cable assumes under its own weight when supported only at its ends. The curve appears as a cross section of the catenoid, the shape assumed by a soap film bounded by two parallel circular rings; this problem is thus often referred to as the soap bubble. The surface of revolution of the catenary curve, the catenoid, is a minimal surface, specifically a minimal surface of revolution, which is what we have just proven above.

Exercise 5.8. *Using the first integral of Euler's equation, show that the extremal of*

$$I[y] = \int_0^1 \frac{\sqrt{1+y'^2}}{y}\,dx, \qquad \begin{matrix} y(0) = 1 \\ y(1) = 1 \end{matrix},$$

satisfies $\pm\int \frac{Cydy}{\sqrt{1-C^2y^2}} = \int dx + A$, *where* A *and* C *are arbitrary constants. Hence show that the general form of the extremals is* $(x+A)^2 + y^2 = \frac{1}{C^2}$ *and use the boundary conditions to show that the extremal is a circle center* $\left(\frac{1}{2}, 0\right)$, *radius* $\frac{\sqrt{5}}{2}$.

5.2.2 Transversality Conditions

We now consider the situation when one, or both, boundary conditions are not provided. The first case we consider is when the end point boundary condition, $y(b)$, is not given. This then makes the functional

$$\min / \max I[y] = \int_a^b F\left(x, y, y'\right) dx, \quad y \in \mathbb{C}^2[a,b], \quad y(x) = \alpha. \tag{5.45}$$

We assume a variation of the form $y = y_0 + \varepsilon\eta$, whereas before $\eta \in \mathbb{C}^2[a,b]$ and $y_0 \in \mathbb{C}^2[a,b]$ is an assumed extremum. The boundary condition at the start point for the admissible variation is $\eta(a) = 0$, but $\eta(b)$ is unspecified. Fig. 5.9 shows an example of the effects not having an end boundary condition has; the three possible solutions y_0, $y_0 + \varepsilon\eta_1$ and $y_0 + \varepsilon\eta_2$, satisfy the initial condition, but all have different end point values.

As in the situation where both boundary conditions are specified, we start by forming the change in the functional, $\Delta I = I[y_0 + \varepsilon\eta] - I[y_0]$:

$$\begin{aligned}
\Delta I &= I[y_0 + \varepsilon\eta] - I[y_0], \\
&= \int_a^b F\left(x, y_0 + \varepsilon\eta, y_0' + \varepsilon\eta'\right) dx - \int_a^b F\left(x, y_0.y_0'\right) dx, \\
&= \varepsilon \int_a^b \left(\eta F_y + \eta' F_{y'}\right) dx + O\left(\varepsilon^2\right), \\
&= \varepsilon\delta I + O\left(\varepsilon^2\right).
\end{aligned} \tag{5.46}$$

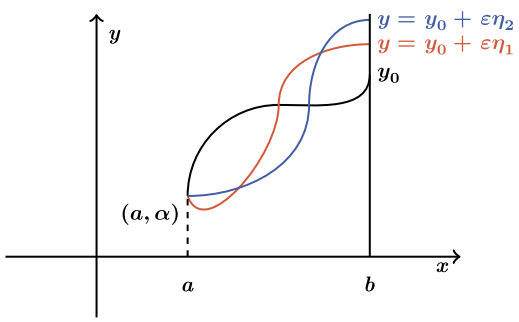


FIGURE 5.9

Illustration of possible solutions when the end boundary condition is not specified.

As before ΔI must be one signed, which implies that δI must equal zero, for sufficiently small ε. The next step is to consider $\delta I = 0$, $\forall \eta \in \mathbb{C}^2[a,b]$, and for $\eta(a) = 0$. Integrating δI from (5.46) by parts we obtain

$$\delta I = \int_a^b \eta \left(F_y - \frac{d}{dx} F_{y'} \right) dx + \eta F_{y'} \Big|_a^b = 0. \tag{5.47}$$

It is required that (5.47) be true for all sufficiently smooth variations $\eta \in \mathbb{C}^2[a,b]$ with the condition $\eta(a) = 0$. From Lagrange's Lemma we know that the Euler equation is equal to zero and we know that the second term in (5.47) is equal to zero at a due to $\eta(a) = 0$. This then leaves

$$\eta(b)\ F_{y'}\Big|^b = 0, \quad \forall \eta \quad \text{and} \quad \eta(b) \neq 0.$$

The condition above leads to what is referred to as the **transversality condition**, which is

$$F_{y'}(x, y_0, y_{0'})\big|_{x=b} = 0 \Rightarrow F_{y'}\left(b, y_0(b), y_0'(b)\right) = 0. \tag{5.48}$$

If the boundary conditions is not given at the starting point a, but is given at the end point b, then the transversality condition becomes $F_{y'}\big|_{x=a} = 0$. If neither boundary conditions are given, then we have to solve two transversality conditions, one at each boundary.

Example 5.9. *Find the extremal of the following functional:*

$$I[y] = \int_0^3 \left(y'^2 - 2yy' + 2ye^x \right) dx, \tag{5.49}$$

where $y(0) = 1$ *but* $y(3)$ *is not given.*

The first step is to form the Euler equation for (5.49), which is

$$\begin{aligned} F_{y_0} - \frac{d}{dx} F_{y_0'} &= -2y_0' + 2e^x - \frac{d}{dx}\left(2y_0' - 2y_0\right) = 0, \\ &= -2y_0'' + 2y_0' - 2y_0' + 2e^x = 0, \\ &\Rightarrow y_0'' = e^x. \end{aligned} \tag{5.50}$$

Therefore, we have to integrate (5.50) twice to obtain the extremal. This can be shown to be

$$y_0 = e^x + Ax + B. \tag{5.51}$$

Given the boundary condition $y_0(0) = 1 \Rightarrow B = 0$, we now use the transversality condition at $x = 3$ to solve for A.

$$\begin{aligned} F_{y'}\big|_{x=b=3} &= 0, \\ \left(2y_0' - 2y_0\right)\big|_{x=3} &= 0, \\ \Rightarrow y_0(3) &= y_0'(3), \\ e^3 + A &= e^3 + 3A \Rightarrow A = 0. \end{aligned}$$

Therefore, the extremal for (5.49) is $y_0 = e^x$.

Exercise 5.10. *Find the extremal of* $I[y] = \int_0^2 y'^2 - 2yy' + 2ydx$ *is*

$$(a)\ y(1) = 1,\ y(2) = 3; \qquad (b)\ y(0) = 1.$$

5.3 Functional With Higher-Order Derivatives

Many problems can arise that involve functionals that contain higher order derivatives higher. We start by considering the general expression for a functional that contains first- and second-order derivatives:

$$I[y] = \int_a^b F\left(x, y, y', y''\right) dx, \quad y \in \mathbb{C}^4[a,b], \quad \begin{array}{c} y(a) = \alpha \\ y(b) = \beta, \\ y'(a) = \gamma \\ y'(b) = \delta \end{array}. \tag{5.52}$$

We require that y be continuous, and differentiable, up to its fourth derivative over the range $[a, b]$.

As with the functionals that only contain the first derivative, we introduce a variation such that $y = y_0 + \varepsilon\eta$ be an admissible variation, whereas before, y_0 is an extremum, so that $\eta \in \mathbb{C}^4[a,b]$ with the conditions that $\eta(a) = \eta(b) = \eta'(a) = \eta'(b) = 0$. The next step is to form the change in the functional $\Delta I = I[y_0 + \varepsilon\eta] - I[y_0]$, which for the second-order derivative case is

$$\Delta I = \int_a^b F\left(x, y_0 + \varepsilon\eta, y_0' + \varepsilon\eta', y_0'' + \varepsilon\eta''\right) dx - \int_a^b F\left(x, y_0, y_0', y_0''\right) dx. \tag{5.53}$$

As for the first-order derivative only case, we expand (5.53) as a Taylor series to the second order, which results in

$$\begin{aligned} \Delta I &= \int_a^b \left(F + \varepsilon\eta F_y + \varepsilon\eta' F_{y'} + \varepsilon\eta'' F_{y''}\right) dx - \int_a^b F dx + O\left(\varepsilon^2\right), \\ &= \varepsilon\delta I + O\left(\varepsilon^2\right). \end{aligned} \tag{5.54}$$

As with the previous sets of functionals, it is required that ΔI be one signed for all admissible η and sufficiently small ε, such that $\delta I = 0$, which implies that

$$\delta I = \int_a^b \left(\eta F_y + \eta' F_{y'} + \eta'' F_{y''}\right) dx = 0. \tag{5.55}$$

Next we apply integration by parts to multiple derivatives of F, which result in

$$\int_a^b \left(\eta F_y - \eta \frac{d}{dx}\left(F_{y'}\right)\right) dx + \eta F_{y'}\Big|_a^b + \eta' F_{y''}\Big|_a^b - \int_a^b \eta' \frac{d}{dx}\left(F_{y''}\right) dx = 0, \tag{5.56}$$

for all admissible η. We now apply integration by parts to the second integral in (5.56) to obtain

$$\begin{aligned} \Delta I &= \int_a^b \left(\eta F_y - \eta \frac{d}{dx}\left(F_{y'}\right)\right) dx + \eta F_{y'}\Big|_a^b + \eta' F_{y''}\Big|_a^b - \eta \frac{d}{dx} F_{y''}\Big|_a^b + \int_a^b \eta \frac{d^2}{dx^2} F_{y''} dx = 0, \\ &= \int_a^b \eta \left(F_y - \eta \frac{d}{dx}\left(F_{y'}\right) + \frac{d^2}{dx^2} F_{y''}\right) dx + \eta\left(F_{y'} - \frac{d}{dx} F_{y''}\right)\Big|_a^b + \eta F_{y''}\Big|_a^b. \end{aligned} \tag{5.57}$$

Since we have the conditions that $\eta(a) = \eta(b) = \eta'(a) = \eta'(b) = 0$, we can apply an extension of Lagrange's lemma to obtain the second-order Euler equation as

$$F_y - \eta \frac{d}{dx}\left(F_{y'}\right) + \frac{d^2}{dx^2} F_{y''} = 0. \tag{5.58}$$

It is possible to extend this result to the nth-order derivative so that for a functional of the form

$$\min / \max \int_a^b F\left(x, y, y', y'', y''', y'''', \ldots, y^{(n)}\right) dx, \tag{5.59}$$

the associated Euler equation is

$$F_y - \frac{d}{dx}F_{y'} - \frac{d^2}{dx^2}F_{y''} + \frac{d^3}{dx^3}F_{y'''} - \frac{d^4}{dx^4}F_{y''''} + \cdots + (-1)^n \frac{d^n}{dx^n}F_{y^{(n)}} = 0. \tag{5.60}$$

Returning to the second-order derivative case, if any of the four boundary conditions are missing, then the associated transversality conditions are

$$\begin{aligned} y(a) &= ? & F_{y'} - \frac{d}{dx}F_{y'}\Big|_a &= 0, \\ y(b) &= ? & F_{y'} - \frac{d}{dx}F_{y'}\Big|_b &= 0, \\ y'(a) &= ? & F_{y''}\big|_a &= 0, \\ y'(b) &= ? & F_{y''}\big|_b &= 0. \end{aligned}$$

5.4 Three-Dimensional Problems

We now consider three-dimensional problems. A curve in three dimensions can be described as a pair of expressions $\begin{matrix} y = y(x) \\ z = z(x) \end{matrix}$. This leads to functionals that contain the first derivatives of y and z as

$$I[y, z] = \int_a^b F\left(x, y, y', z, z'\right) dx, \quad \begin{matrix} y(a) = \alpha \\ z(a) = \beta \\ y(b) = \gamma \\ z(b) = \delta \end{matrix}. \tag{5.61}$$

An illustration of a curve in 3D is shown in Fig. 5.10. For the types of functionals in (5.61), we have to introduce two admissible variations: $\eta_1(x)$ for y and $\eta_2(x)$ for z. We also require two sufficiently small constants, ε_1 and ε_2, for y and z, respectively. Therefore, we have a y extremum, y_0, and a z extremum, z_0, such that $y = y_0 + \varepsilon_1\eta_1$ and $z = z_0 + \varepsilon_2\eta_2$, with the standard boundary conditions: $\eta_1(a) = \eta_1(b) = \eta_2(a) = \eta_2(b) = 0$. The next step is to consider the change in the functional, ΔI, which for the three-dimensional problem is

$$\begin{aligned} \Delta I &= I\left[y_0 + \varepsilon_1\eta_1, y_0' + \varepsilon_1\eta_1', z_0 + \varepsilon_2\eta_2, z_0' + \varepsilon_2\eta_2'\right] - I[y_0 z_0], \\ &= \varepsilon_1 \int_a^b \eta_1 \left(F_{y_0} - \frac{d}{dx}F_{y_0'}\right) dx + \varepsilon_1\eta_1 F_{y_0}\big|_a^b \\ &\quad + \varepsilon_2 \int_a^b \eta_2 \left(F_{z_0} - \frac{d}{dx}F_{z_0'}\right) dx + \varepsilon_2\eta_2 F_{z_0}\big|_a^b + O\left(\varepsilon^2\right). \end{aligned} \tag{5.62}$$

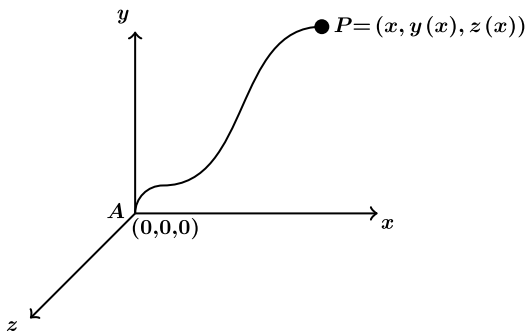


FIGURE 5.10

Illustration of the three-dimensional calculus of a variation problem.

Given boundary conditions for η_1 and η_2, we know that the two non-integral-based expressions in (5.62) are zero. Therefore, applying Lagrange's Lemma to the two integrals in (5.62) results in a coupled nonlinear system of partial differential equations for the Euler equation for y and z as

$$F_{y_0'} - \frac{d}{dx} F_{y_0'} = 0, \tag{5.63}$$

$$F_{z_0'} - \frac{d}{dx} F_{z_0'} = 0. \tag{5.64}$$

Example 5.11. *Find the extremals that start at* $A\,(0, 0, 0)$ *and end at* $B\,(1, 1, 1)$ *for the functional*

$$I[y, z] = \int_0^1 \frac{\sqrt{1 + y'^2 + z'^2}}{x} dx.$$

The starting point is to form the Euler equations for y and z, which are

$$0 - \frac{d}{dx}\left(\frac{y_0'}{\sqrt{1 + y_0'^2 + z_0'^2}\, x}\right) = 0,$$

$$0 - \frac{d}{dx}\left(\frac{z_0'}{\sqrt{1 + y_0'^2 + z_0'^2}\, x}\right) = 0.$$

Integrating both sides of the differential equations above introduces the two constants C and D, such that after some simple rearranging we obtain

$$y_0' = Cx\sqrt{1 + y_0'^2 + z_0'^2}, \tag{5.65}$$

$$z_0' = Dx\sqrt{1 + y_0'^2 + z_0'^2}. \tag{5.66}$$

Dividing (5.65) by (5.66) results in

$$\frac{y_0'}{z_0'} = \frac{C}{D}.$$

We have to consider if it is possible that $D = 0$. If $D = 0$, then that would imply from (5.66) that $z_0' = 0$, which would make z_0 a constant. However, we have boundary conditions that when $x = 0$ then $z = 0$ and at $x = 1$, then $z = 1$. Therefore, it is not possible for a constant to satisfy both boundary conditions; this then implies that $D \neq 0$. The same argument can be applied to C as well.

With this new information about C and D, it is possible to integrate the ratio above so that

$$y_0(x) = Pz_x + Q \qquad \begin{matrix} y_0(0) = 0 \\ z_0(0) = 0 \end{matrix} \Rightarrow Q = 0,$$

$$y_0(x) = Pz(x) \qquad \begin{matrix} y_0(1) = 1 \\ z_0(1) = 1 \end{matrix} \Rightarrow P = 1.$$

This implies that $y_0(x) = z_0(x)$.

To find a solution to the coupled nonlinear system of partial differential equations in (5.65) and (5.66), we return to (5.65). Squaring both sides of (5.65) and using the fact that $z_0 = y_0$ results in

$$\begin{aligned} y_0'^2 &= C^2x^2\left(1 + 2y'^2\right), \\ \left(1 - 2C^2x^2\right)y'^2 &= C^2x^2, \\ \Rightarrow \frac{dy}{dx} &= \pm\frac{Cx}{\sqrt{1 - 2C^2x^2}}, \\ y_0 &= \pm\int \frac{Cx}{\sqrt{1 - 2C^2x^2}} + E, \\ y_0 &= \mp\frac{\sqrt{1 - 2C^2x^2}}{2C} + E. \end{aligned} \tag{5.67}$$

To remove the $\mp$ sign, we square (5.67) and rearrange so that

$$\begin{aligned} (y_0 - E)^2 &= \frac{1 - 2C^2x^2}{4C^2}, \\ \frac{x^2}{2} + (y_0 - E)^2 &= \frac{1}{4C^2}. \end{aligned}$$

The next step is to evaluate the boundary conditions: $y_0(0) = 0$ and $y_0(1) = 1$. This implies that $\frac{1}{4C^2} = E^2$ from the first boundary condition, and from the second boundary condition $E = \frac{3}{4}$. This results in

$$\frac{x^2}{2} + \left(y_0 - \frac{3}{4}\right)^2 = \frac{9}{16},$$

which is the equation for an ellipse. However, as $y_0 = z_0$ we have

$$\frac{x^2}{2} + \left(z_0 - \frac{3}{4}\right)^2 = \frac{9}{16},$$

and as such this is an elliptical cylinder.

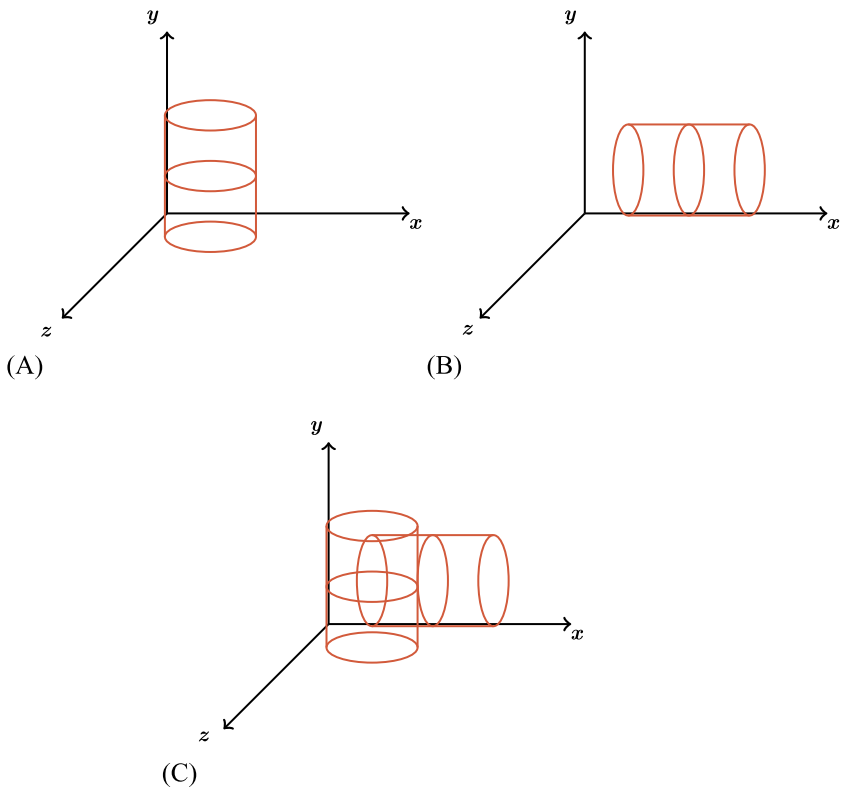


FIGURE 5.11

Illustration of the two elliptical cylinders in the top two plots; the intersection of the two cylinders, shown in the bottom plot, is the solution.

Applying the same argument derived above for $z_0 = y_0$, we then have y_0 free. This gives another elliptical cylinder that is perpendicular to the first one (see Fig. 5.11 for an illustration). As both cylinders must hold, this implies that the extremal is the curve of the intersection of the two perpendicular elliptical cylinders (again see Fig. 5.11 for an illustration).

5.5 Functionals With Constraints

To illustrate how to solve the class of problems where the functional has a constraint, we are going to solve Dido's problem from Section 5.1.4. To recap, Dido's problem is where we have a length of wire of length $2L$ that is attached to the points $A\,(-a, 0)$ and $A\,(a, 0)$ on the x-axis and we have to find the shape of the wire, $y\,(x)$, such that the area enclosed above the x-axis is maximized.

The historical contents of the name of this problem comes from its association with Queen Dido, who was originally a princess of Tyre. Her brother, the King, had her husband killed and so she fled

when he would not share the throne with her. She travelled to Northern Africa, to what is now Tunisia, where she offered to buy land from King Jurbas of Libya. She was allowed as much land as she could enclose with an oxen's hide. The clever part is that she had the hide cut into almost infinitesimally small strips and then attached these strips together; she then used the much longer strip to enclose the maximum area possible. The sea was used as the x-plane boundary as we have mathematically described the problem. The enclosed area became the city of Carthage in Tunisia and Princess Dido became the first queen of Carthage. Parts of Carthage can still be found in the modern-day city of Tunis.

The mathematical starting point is to consider the equation for area in terms of integrating under the curve. Therefore, the function is $I[y] = \int_{-a}^{a} y dx$ subject to the constraint that the length of the wire is $2L$. We have already used the functional for distance earlier. We therefore have a second functional $J[y]$, which is $J[y] = \int_{-a}^{a} \sqrt{1 + y'^2} dx = 2L$, subject to $y(-a) = y(a) = 0$. We also have the boundary conditions that $2L > 2a$, where a is the distance from the origin to point A, and $2L < \pi a$, which is the circumference of a semicircle.

As mentioned in the title of this section, this is the case where we have a functional with a constraint. As in non-functional-based minimize/maximize problems where there is a constraint, we also introduce a Lagrange multiplier to help solve the problem. Therefore, we have a third functional, $K[y] \equiv I[y] + \lambda J[y]$, where λ is the Lagrange multiplier. This now makes the problem

$$K[y] = \int_{-a}^{a} y + \lambda \sqrt{1 + y'^2} dx. \tag{5.68}$$

As for all the previous cases, the next step is to form the Euler equation, which for (5.68) is

$$1 - \frac{d}{dx}\left(\frac{\lambda y_0'}{\sqrt{1 + y_0'^2}}\right) = 0. \tag{5.69}$$

Rearranging, and integrating (5.69), results in

$$\frac{\lambda y_0'}{\sqrt{1 + y_0'^2}} = x + B. \tag{5.70}$$

The next step are: square both sides of (5.70), multiply throughout by $\left(1 + y_0'^2\right)$, factorize $y_0'^2$, divide by the factors of $y_0'^2$, and then take the square root. Following these steps results

$$y_0' = \frac{\pm (x + B)}{\sqrt{\lambda^2 - (x - B)^2}}. \tag{5.71}$$

Next, we integrate (5.71) so that we remove the derivative off of y_0. Therefore, we have

$$\begin{aligned} y_0 &= \pm \int \frac{x + B}{\sqrt{\lambda^2 - (x + B)^2}} dx + C, \\ &= \mp \sqrt{\lambda^2 - (x + B)^2} + C, \\ \lambda^2 &= (x + B)^2 + (y_0 - C)^2. \end{aligned} \tag{5.72}$$

Therefore, (5.72) is the equation for a circle centered at $(-B, C)$ with radius $r = \lambda$.

The next step is to consider the two boundary conditions: $y(a) = y(-a) = 0$. This leads to the pair of equations

$$(a + B)^2 + C^2 = \lambda^2, \tag{5.73}$$

$$(-a + B)^2 + C^2 = \lambda^2. \tag{5.74}$$

Subtracting (5.74) from (5.73) results in $4aB = 0$. As we know a is not zero, this implies that $B = 0$. This then enables us to find an expression for C that is $C = \pm\sqrt{\lambda^2 - a^2}$. This means that we have a circle: $x^2 + (y_0 - C)^2 = \lambda^2$ that is centered at $(0, C)$ with radius λ. But we now have to consider the $\pm$ in front of the square root and decide if the center of the circle is a positive or negative square root.

We consider the positive square root first. This makes $C = \sqrt{\lambda^2 - a^2} > 0$; however, this would then make the path drawn out by the wire larger than a semicircle, and we also have the boundary condition constraint, $2L < \pi a$. Therefore, the negative center is the solution to consider, so $C = -\sqrt{\lambda^2 - a^2}$. As we can see from Fig. 5.12, having the center of the circle below the x-axis creates a shape that appears as a triangle with a semicircle on top. This implies that to find the total area constrained by the wire above the x-axis, we will need to calculate the total area of the quadrant and then subtract the area below the x-axis. As shown in Fig. 5.12, we have an angle, α, that enables the circular part of the solution to pass through a and $-a$ on the x-axis, so that all the other boundary conditions are still satisfied. Therefore, the maximum area is

$$\max A = \frac{1}{2}\lambda^2 2\alpha - 2\frac{1}{2}a\lambda\cos\alpha = \lambda^2\alpha - a\lambda\cos\alpha.$$

We also have the $2L$ distance constraint, which implies that $2L = \lambda 2\alpha \Rightarrow \lambda = \frac{L}{\alpha}$. Next we substitute this information into the area equation above and obtain

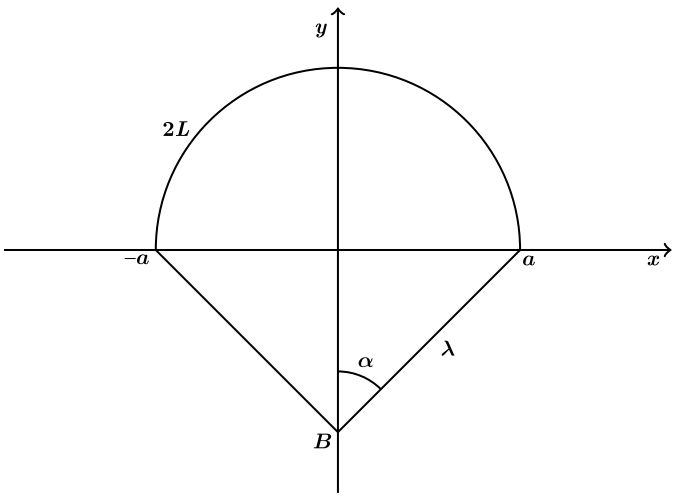


FIGURE 5.12

Illustration of the arc length solution to the Dido maximum enclosed area.

$$A = \frac{L^2}{\alpha} - \frac{aL}{\alpha}\cos\alpha.$$

We also have the property that $\sin\alpha = \frac{a}{\lambda} = \frac{a}{\frac{L}{\alpha}} = \frac{a}{L}\alpha$, which is equivalent to $\sin\alpha = k\alpha$. The next step is to ensure that the angle α still enables us to satisfy the boundary constraints. The angle cannot be greater than $\frac{\pi}{2}$ as $\cos\frac{\pi}{2} = 0$. Comparing the gradients of $\sin\alpha = k\alpha$, we have that $\cos\alpha = k$ and that we consider values of α where the straight line $k\alpha$ is equal to $\sin\alpha$. Therefore, we must have that $k < 1$, which implies that $\frac{a}{L} < 1$ and we have that $k > \frac{\sin\frac{\pi}{2} - \sin 0}{\frac{\pi}{2} - 0} \Rightarrow k > \frac{2}{\pi} \Rightarrow \frac{a}{L} > \frac{2}{\pi}$.

Another way to think of this condition is to consider that if $\alpha = \frac{\pi}{2}$, then $\sin\frac{\pi}{2} = 1$. However, if we consider the gradient of $\sin\alpha = k\alpha$, then at $\frac{\pi}{2}$ we have $\cos\frac{\pi}{2}$ which is zero. This would then imply that $k = 0$, which means that $\frac{a}{L} = 0$, which cannot be. Therefore, the angle α must be less than $\frac{\pi}{2}$ but also $\frac{a}{L} < \frac{2}{\pi}$.

To illustrate the Dido type problem, maximum enclosed area, we consider the following example:

Example 5.12. *Show that the maximum value of* $I[y] = \int_{-1}^{1} y dx$, *with the boundary conditions* $y(-1) = y(1) = 0$ *subject to the constraint* $\int_{-1}^{1}\sqrt{1+y'^2}dx = \frac{2\pi}{3}$ *is* $\frac{2\pi}{3} - \sqrt{3}$.

Solution. This problem is equivalent to saying find the maximum area possible under the curve of fixed length $\frac{2\pi}{3}$ between the points $(-1, 0)$ and $(1, 0)$.

Therefore, as we derived above, this is a constraint problem, so using a Lagrange multiplier-based approach, we have the new functional

$$J[y] = \int_{-1}^{1} y + \lambda\sqrt{1+y'^2}dx,$$

where λ is a constant.

Forming the Euler equation for this problem results in

$$1 - \frac{d}{dx}\frac{\lambda y'}{\sqrt{1+y'^2}} = 0 \Rightarrow \frac{\lambda y'}{\sqrt{1+y'^2}} = x + A. \tag{5.75}$$

Rearranging the square of the right equation in (5.75), and taking the square root, results in the integrable expression for y' as

$$y' = \pm\frac{x+A}{\sqrt{\lambda^2 - (x+A)^2}} \Rightarrow y = \mp\int\frac{x+A}{\sqrt{\lambda^2-(x+A)^2}}dx + B = \mp\sqrt{\lambda^2-(x+A)^2} + B. \tag{5.76}$$

As we have seen above, rearranging (5.76) results in the equation for a circle, $(y-B)^2 + (x+A)^2 = \lambda^2$, where the circle is centered at $(-A, B)$ and has a radius $= \lambda$. Applying the two boundary conditions results in $A = 0$ and $B = \pm\sqrt{r^2 - 1}$. Therefore, our solution is a circle centered at $(0, B)$ with radius r.

The next step is to determine if the center of the circle is greater than $y = 0$ of less than. In Fig. 5.13 we have drawn the two solutions for this problem. Fig. 5.13A shows the solution for $B > 0$, while Fig. 5.13B presents the solution for $B < 0$.

If $B < 0$ then the length of the arc of the circle is less than the length of a semicircle, i.e., the solution if $B = 0$, where the circumference of a circle is given by $d = 2\pi r$, but for a semicircle it is half of this equation and for the current formulation the radius at $B = 0$ is 1. Therefore, the semicircle has

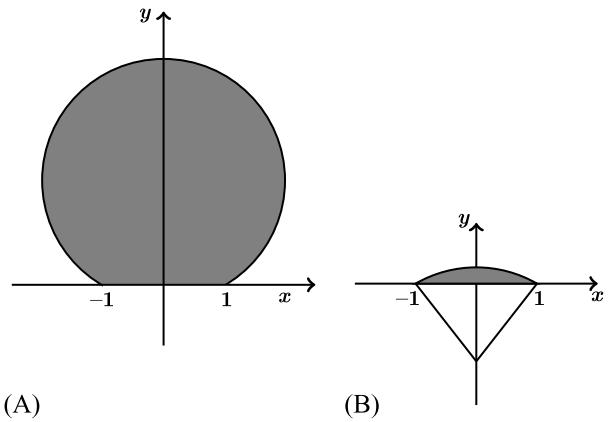


FIGURE 5.13

Illustration of the area enclosed by (A) for $B > 0$ and (B) for $B < 0$ in the Dido example.

a boundary length of π. Thus, given the constraint that the length of the bounding wire cannot be more than $\frac{2\pi}{3}$, we have the situation that if $B > 0$, then the circle centered there would have a length larger than the semicircle's, which is already greater than the constraint. This implies that $B < 0$.

Given that we must use the negative value for B, we now have the arc of the circle as $\left(y + \sqrt{r^2 - 1}\right)^2 + x^2 = r^2$. From Fig. 5.13 we can see that the length of the arc is $2r\alpha$, but that this must equal $\frac{2\pi}{3}$. This implies that $r = \frac{\pi}{3\alpha}$. From the diagram we can see that we also have, through the trigonometric identities, that $r \sin\alpha = 1$. Therefore, $\sin\alpha = \frac{1}{r} = \frac{3\alpha}{\pi}$ which occurs at $\alpha = \frac{\pi}{6}$, so that $\sin\frac{\pi}{6} = \frac{3\frac{\pi}{6}}{\pi} = \frac{1}{2}$, which makes $r = 2$. This now lets us know that the circle is centered at $\left(0, -\sqrt{3}\right)$.

To find the maximum area enclosed by the wire at this point, we take calculate the area of the sector of the circle enclosed between $(-1, 0)$ and $(1, 0)$ and subtract the area of the two triangles below the x-axis. Therefore, we have that the total area enclosed by the curve is $2\frac{1}{2}r^2\alpha - 2\frac{1}{2}r\cos\alpha 1 = \frac{2^2\pi}{6} - 2\cos\frac{\pi}{6} = \frac{2\pi}{3} - \sqrt{3}$, which is the answer we were seeking.

5.6 Functional With Extremals That Are Functions of Two or More Variables

The type of problems that involve functions of more than one variable are describing a surface, as shown in Fig. 5.14. The general functional for this type of problem is of the form

$$I[u] = \iint_D F\left(x, y, u, u_x, u_y\right) dx\, dy, \tag{5.77}$$

where F is given and $u(x, y)$ on the boundary of D, called C.

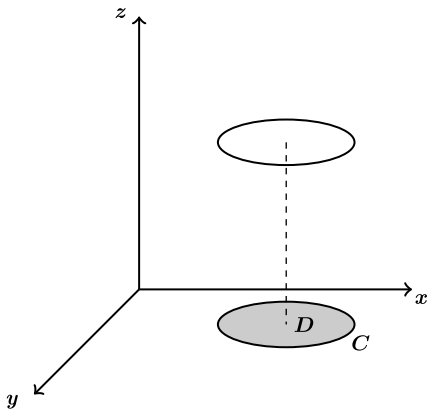


FIGURE 5.14

Illustration of the surface that has to be integrated over to find the extremum.

We begin with a simple example:

$$I[u] = \iint_D u_x^2 + u_y^2 + 2g(u)\,dx\,dy, \tag{5.78}$$

and derive the Euler equation from first principles. We start by assuming that $u_0(x, y)$ is an extremum of (5.78), and that $u(x, y) = u_0(x, y) + \varepsilon\eta(x, y)$ is an admissible variation, with $\eta = 0$ on the boundary C, since $u = u_0 = f$ on the boundary C. Forming the change in I, $\delta I = I[u_0 + \varepsilon\eta] - I[u_0]$, we obtain

$$\begin{aligned}
\delta I &= I[u_0 + \varepsilon\eta] - I[u_0] \\
&= \iint_D \left(u_{0,x} + \varepsilon\eta_x\right)^2 + \left(u_{0,y} + \varepsilon\eta_y\right)^2 + 2g(u_0 + \varepsilon\eta)\,dx\,dy - \iint_D u_x^2 + u_y^2 + 2g(u)\,dx\,dy, \\
&= \iint_D 2\varepsilon u_{0,x}\eta_x + 2\varepsilon u_{0,y}\eta_y + 2\varepsilon\eta g'(u_0)\,dx\,dy, \qquad (5.79) \\
&= \varepsilon^2 \iint_D \eta_x^2 + \eta_y^2 + eta^2 g''(u_0)\,dx\,dy + O\left(\varepsilon^3\right), \qquad (5.80) \\
&= \varepsilon\delta I + O\left(\varepsilon^2\right).
\end{aligned}$$

We require, as before, $\delta I = 0$ as ΔI is one signed. Therefore, for all admissible variations, η, we require

$$\iint_D \varepsilon u_{0,x}\eta_x + \varepsilon u_{0,y}\eta_y + \varepsilon\eta g'(u_0)\,dx\,dy = 0. \tag{5.81}$$

In order to solve (5.81) we have to integrate by parts; however, for the type of problem we have here we cannot use standard integration by parts. We now introduce **Green's theorem**, which will enable us to solve (5.81).

Theorem 5.13. *Let C be a positively orientated, piecewise smooth, simple closed curve in a plane, and let D be the region bounded by C. If P and Q are functions of (x, y) defined on an open region*

containing D and have continuous partial derivatives there, then

$$\iint_D \left(\frac{\partial P}{\partial x} - \frac{\partial Q}{\partial y} \right) dx\, dy = \oint_C (Q\, dx + P\, dy), \tag{5.82}$$

where the path of integration along C is counterclockwise.

Returning to our example, we first note that

$$u_{0,x}\eta_x = \left(u_{0,x}\eta\right)_x - u_{0,xx}\eta, \tag{5.83}$$

$$u_{0,y}\eta_y = \left(u_{0,y}\eta\right)_y - u_{0,yy}\eta, \tag{5.84}$$

which gives us

$$0 = \iint_D \frac{\partial}{\partial x}\left(u_{0,x}\eta\right) - \eta u_{0,xx} + \frac{\partial}{\partial y}\left(u_{0,y}\eta\right) - \eta u_{0,yy} + \eta g'(u_0)\, dx\, dy. \tag{5.85}$$

By Green's theorem we have $P = \left(u_{0,x}\eta\right)_x$ and $Q = \left(u_{0,y}\eta\right)_y$, which means we can rewrite (5.85) as

$$\iint_D \left(u_{0,x}\eta\right)_x + \left(u_{0,y}\eta\right)_y dx\, dy = \oint_C u_{0,x}\eta\, dy - \oint_C u_{0,y}\eta\, dx = 0, \tag{5.86}$$

due to $\eta = 0$ on the boundary C. Therefore, the remaining terms from (5.81) are

$$\iint_D \eta\left(g'(u_0) - u_{0,xx} - u_{0,yy}\right) d\, dy = 0. \tag{5.87}$$

Through a further extension of Lagrange's lemma, it is possible to infer that

$$\frac{\partial^2 u_0}{\partial x^2} + \frac{\partial^2 u_0}{\partial y^2} = g'(u_0). \tag{5.88}$$

Therefore, (5.88) is the Euler equation that needs to be solved to find the extremum for these types of problems. Note that the type of second-order partial differential equation in (5.88) is referred to a **Poisson equation**.

Example 5.14. *Find the extremal of the following functional:*

$$I[u] = \iint_D u_x^2 + u_y^2 + 2udx\, dy, \tag{5.89}$$

where $D = \left\{(x, y) : x^2 + y^2 \leq 1\right\}$ (unit disc) and where $u = 1$ on $C = \left\{(x, y) : x^2 + y^2 = 1\right\}$ (unit circle), and show that the extremal is an extremum.

We have illustrated the domain in Fig. 5.15.

The first step is to form the Euler equation for (5.89);

$$u_{0,xx} + 0_{,yy} = 1 \equiv \nabla^2 u_0 = 1, \tag{5.90}$$

which is a symmetric partial differential equation and where ∇^2 is the Laplacian operator that is defined as

$$\nabla^2 \equiv \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}. \tag{5.91}$$

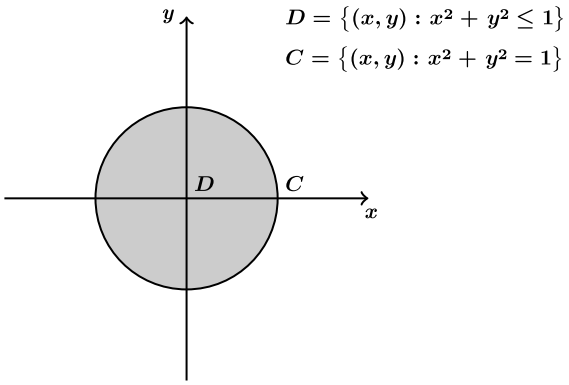


FIGURE 5.15

Illustration of the unit disc domain with the unit circle boundary.

To solve (5.90) on the specified domain, we introduce a solution that only depends on the distance from the origin, which is referred to as r. Therefore, converting (5.90) into polar coordinate (r, θ), where $r^2 = x^2 + y^2 \Rightarrow r = \sqrt{x^2 + y^2}$, we have that $u(x, y) = U(r)$. Next, converting the Laplacian operator into polar coordinates results in

$$\nabla^2 u_0 = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_0}{\partial r}\right) + \frac{\partial^2 u}{\partial \theta^2} = 1.$$

The boundary condition becomes C for $r = 1$ and the domain becomes D for $r \leq 1$.

We now consider how to convert $u_{xx} + u_{yy} = 1$ to be in terms of U. We start with

$$\frac{\partial u}{\partial x} = \frac{\partial r}{\partial x}\frac{\partial U}{\partial r} = r_x \frac{dU}{dr}.$$

Using the definition for r^2, we can obtain an expression for r_x as $r_x = \frac{x}{r}$. Therefore, we can write

$$\frac{\partial}{\partial x} = \frac{x}{r}\frac{d}{dr}. \tag{5.92}$$

Applying this derivation again, we can obtain an expression for the second derivative operator in terms of polar coordinates as

$$u_{xx} = \frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x}u_x = \frac{\partial}{\partial x}\left(\frac{x}{r}\frac{dU}{dr}\right).$$

Applying the product rule and substituting (5.92), we obtain

$$\frac{\partial}{\partial x}\left(x\frac{1}{r}\frac{dU}{dr}\right) = \frac{1}{r}\frac{dU}{dr} + x\frac{\partial}{\partial x}\left(\frac{1}{r}\frac{dU}{dr}\right). \tag{5.93}$$

Substituting (5.92) into (5.93) yields

$$\frac{\partial}{\partial x}\left(x\frac{1}{r}\frac{dU}{dr}\right)=\frac{1}{r}\frac{dU}{dr}+\frac{x^2}{r}\frac{d}{dr}\left(\frac{1}{r}\frac{dU}{dr}\right).$$

This implies that $u_{0,xx}$, and by association, $u_{0,yy}$ convert to

$$u_{0,xx}\rightarrow\frac{1}{r}\frac{dU}{dr}+\frac{x^2}{r}\frac{d}{dr}\left(\frac{1}{r}\frac{dU}{dr}\right), \tag{5.94}$$

$$u_{0,yy}\rightarrow\frac{1}{r}\frac{dU}{dr}+\frac{y^2}{r}\frac{d}{dr}\left(\frac{1}{r}\frac{dU}{dr}\right). \tag{5.95}$$

Therefore, adding (5.94) and (5.95) enables the original partial differential equation to be written in polar coordinates as

$$u_{0,xx}+u_{0,yy}\equiv\frac{2}{r}\frac{dU}{dr}+r\frac{d}{dr}\left(\frac{1}{r}\frac{dU}{dr}\right). \tag{5.96}$$

Expanding the differential operator using the product rule results in

$$u_{0,xx}+u_{0,yy}=1\rightarrow\frac{1}{r}\frac{dU}{dr}+\frac{d^2U}{dr^2}=1\Rightarrow\frac{1}{r}\frac{d}{dr}\left(rU'\right)=1. \tag{5.97}$$

We have now arrived at an expression that is easier to integrate to find the extremum. Solving for r, we have

$$\begin{aligned}r\frac{dU}{dr}&=\frac{r^2}{2}+A,\\ \Rightarrow\frac{dU}{dr}&=\frac{r}{2}+\frac{A}{r},\\ \Rightarrow U(r)&=\frac{r^2}{4}+A\ln r+B.\end{aligned} \tag{5.98}$$

Using the boundary condition that $U=u=1$ on $r=1$ implies that $B=\frac{3}{4}$. To find the value for A, we have to consider that fact that D contains $r=0$ and that the solution is bounded in D, but U would be unbounded at $r=0$ unless $A=0$. Therefore,

$$U(r)=\frac{r^2}{4}+\frac{3}{4}\Rightarrow u(x,y)=\frac{x^2+y^2}{4}+\frac{3}{4}, \tag{5.99}$$

which is an extremal. The final step is to see if the solution is an extremum. To verify if the solution is an extremum, we consider the second variation $\delta^2 I$, which is

$$I[u_0+\varepsilon\eta]-I[u_0]=\varepsilon^2\iint_D\eta_x^2+\eta_y^2dx\,dy\geq 0, \tag{5.100}$$

for all ε, therefore the solution is both a local and a global minimum.

5.6.1 Three-Dimensional Problems

For three-dimensional problems we are integrating over a volume, V, and have a surface S that has boundary conditions associated with them. Therefore, an integral for a three-dimensional functional would be of the form

$$I[u] \iiint_V F\left(x, y, z, u, {}_x, u_y, u_z\right) dx\, dy\, dz, \tag{5.101}$$

where $u = u(x, y, a)$ and $u = g(x, y, z)$ and the surface S. We shall derive the proof of the Euler equation for this formulation of the functional after we have presented some examples. Therefore, the associated Euler equation for (5.101) is

$$F_u - \frac{\partial}{\partial x} F_{u_x} - \frac{\partial}{\partial y} F_{u_y} - \frac{\partial}{\partial z} F_{u_z} = 0. \tag{5.102}$$

Example 5.15. *Find the extremal of the three-dimensional functional*

$$I[u] = \iiint_v u_x^2 + u_y^2 + u_z^2 + 2g(u)\, dx\, dy\, dz. \tag{5.103}$$

The associated Euler equation for (5.103) is

$$u_{xx} + u_{yy} + u_{zz} = g'(u). \tag{5.104}$$

We consider three different functions that $g(u)$ could be, e.g., $g(u) = c$, where c is a constant that can include the value zero, $g(u) = u$ or $g(u) = u^2$.

If, for example, we were given that $V = \left\{(x, y, z) : x^2 + y + z^2 \leq 1\right\}$ (this is the definition for the unit ball, with the surface $S = \{(x, y, z) \leq 1\}$ where $u(x, y, z) = 1$ on S), then this problem would be symmetric and so implies using the change of variable into polar coordinates again, but now we have $r^2 = x^2 + y^2 + z^2$ so that $u(x, y, z) = U(r)$. By following the same arguments for the polar coordinate change for the two-dimensional problem, it is possible to show that

$$u_{xx} = \frac{x^2}{r^2} u'' + \frac{u'}{r} - \frac{x^2}{r} u'. \tag{5.105}$$

Applying the same arguments to u_{yy} and u_{zz} implies that

$$u_{xx} + u_{yy} + u_{zz} = U'' + 3\frac{U'}{r} - \frac{1}{r} U' = U'' + \frac{2}{r} U' \equiv \frac{1}{r^2} \frac{d}{dr}\left(r^2 \frac{dU}{dr}\right). \tag{5.106}$$

We now consider the case when $g(u) = 0$. This makes the polar coordinate equivalent of the Euler equation as

$$\begin{aligned} \frac{1}{r^2} \frac{d}{dr}\left(r^2 \frac{dU}{dU} dr\right) &= 0, \\ \Rightarrow r^2 \frac{dU}{dr} &= A, \\ \Rightarrow \frac{dU}{dr} = \frac{A}{r^2} &\Rightarrow U = -\frac{A}{r} + B. \end{aligned} \tag{5.107}$$

We have the condition that $u = U = 1$ on $r = 1$, which implies $1 = A + B$; however, we have that $r = 0$ is inside the domain, and the condition that the solution is supposed to be bounded in the domain, therefore, $A = 0$, and then $B = 1$.

We next consider the case where $g(u) = u \Rightarrow g'(u) = 1$. Therefore, $u_{xx} + u_{yy} + u_{zz} = 1$ where in polar coordinates this partial differential equation becomes

$$\begin{aligned}\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{dU}{dr}\right) &= 1,\\ r^2\frac{dU}{dr} &= \frac{r^3}{3} + A,\\ \frac{dU}{dr} &= \frac{r}{3} + \frac{A}{r^2},\\ U &= \frac{r^2}{3} - \frac{A}{r} + B.\end{aligned} \tag{5.108}$$

Using the boundary conditions that $u = U = 1$ on $r = 1$, which gives $1 = \frac{1}{6} - A + B$, but as before the domain contains $r = 0$, which again implies that $A = 0$, therefore we have $1 = \frac{1}{6} + B \Rightarrow B = \frac{5}{6}$. This then makes the solution $U(r) = \frac{r^2}{6} + \frac{5}{6} \Rightarrow u(x, y, z) = \frac{x^2+y^2+z^2}{6} + \frac{5}{6}$.

Finally, we consider the case where $g(u) = u^2$, which makes the polar coordinate-based Euler equation

$$\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{dU}{dr}\right) = 2u. \tag{5.109}$$

We introduce the change of variable

$$U = \frac{V}{r} \Rightarrow \frac{dU}{dr} = \frac{d}{dr}\left(\frac{V}{r}\right) = \frac{1}{r}\frac{dV}{dr} = \frac{1}{r}\frac{dV}{dr} - \frac{1}{r^2}V. \tag{5.110}$$

This then makes (5.109), in terms of V, as

$$\begin{aligned}\frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{dU}{dr}\right) = 2U \rightarrow \frac{1}{r^2}\frac{d}{dr}\left(r\frac{dV}{dr} - V\right) &= \frac{2V}{r},\\ \Rightarrow \frac{1}{r^2}\left(r\frac{d^2V}{dr^2} + \frac{dV}{dr} - \frac{dV}{dr}\right) &= \frac{2V}{r},\\ \frac{d^2V}{dr^2} &= 2V.\end{aligned} \tag{5.111}$$

Therefore, the solution to the type of differential equation in (5.111) is $V(r) = Ae^{\sqrt{2}r} + Be^{\sqrt{2}r} \equiv A'\cosh\sqrt{2}r + B'\sinh\sqrt{2}r$. Substituting for V, we obtain the solution for U as

$$U(r) = \frac{V}{r} = \frac{A'\cosh\sqrt{2}r + B'\sinh\sqrt{2}r}{r}. \tag{5.112}$$

As with the other two situations, we have $r = 0$ included in the domain. Therefore, we must have $A' = 0$ as $\cosh 0 = 1$ but $\sinh 0 = 0$, and therefore we would be dividing by zero if $A' \neq 0$. This then leaves B' to satisfy the boundary condition.

Finally, we return to the Euler equation for the three-dimensional functional and now we perform the integration by parts to separate out η. We start by considering the first variation, which is given by

$$\delta I = \iiint_V \eta F_u + \eta_x F_{u_x} + \eta_y F_{u_y} + \eta_z F_{u_z} dx\, dy\, dz = 0. \tag{5.113}$$

We know that $\eta_x F_{u_x} = \frac{\partial}{\partial x}\left(\eta F_{u_x}\right) - \eta \frac{\partial}{\partial x} F_{u_x}$, which then allows us to write (5.113) as

$$\begin{aligned} 0 = &\iiint_v \eta \left(F_u - \frac{\partial}{\partial x} F_{u_x} - \frac{\partial}{\partial y} F_{u_y} - \frac{\partial}{\partial z} F_{u_z} \right) dx\, dy\, dz \\ &+ \iiint_v \frac{\partial}{\partial x}\left(\eta F_{u_x}\right) + \frac{\partial}{\partial y}\left(\eta F_{u_y}\right) + \frac{\partial}{\partial z}\left(\eta F_{u_z}\right) dx\, dy\, dz. \end{aligned} \tag{5.114}$$

The second term in (5.114) is zero, hence the result from earlier; however, this is because

$$\iiint_V P_x + q_y + R_z dx\, dy\, dz = \iint_s l P + m Q + n R\, dS, \tag{5.115}$$

where $\begin{pmatrix} l \\ m \\ n \end{pmatrix} = \boldsymbol{n}$ is a unit outward normal from S. The equation in (5.115) is that of the **divergence theorem**. For our situation we have $P = \eta F_{u_x}$, $Q = \eta F_{u_y}$ and $R = \eta F_{u_z}$, which means that we are evaluating η on the surface S which is required to be zero. This leads to the remaining term having to be zero, and so his is the Euler equation we stated earlier for the three-dimensional problem.

5.7 Summary

In this chapter we have introduced the theory of calculus of variation and applied it to different types of problems. We have introduced the notion of *taking the first variation* of a functional, as well as the Euler equation; you may see these equations referred to as Euler-Lagrange equations. We have illustrated how the theory can be extended to multiple dimensions and functions of multiple variables. The reason for introducing the topic of calculus of variation is to show the theory that puts the VAR into the variational data assimilation systems. Calculus of variation also plays an important role in the theory of representer-based data assimilation/prediction, and in optimal control theory, where control theory plays an important part in the derivation of the Kalman filter. Therefore, we now move on to control theory.

CHAPTER

Introduction to Control Theory

6

Contents

6.1 The Control Problem

The starting point for control theory is to consider a system of linear ordinary differential equations, for example;

$$\begin{aligned} \dot{x}_1 &= 5x_1 - 2x_2, \\ \dot{x}_2 &= 7x_1 - 4x_2, \end{aligned} \quad \equiv \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}, \tag{6.1}$$

where $x_1 \equiv x_1(t)$ and $x_2 \equiv x_2(t)$, and t is time. Therefore, we are seeking solutions to (6.1). The superscript dot, $\dot{}$, is referring to the derivative of x_1 and x_2 with respect to time, i.e., $\dot{} \equiv \frac{d}{dt}$. While we have not shown how to solve this type of linear system of ordinary differential equations in this book, it can be found in most good introduction to calculus textbook. However, the technique that we use here involves eigenvalues and eigenvectors, which we did introduce earlier.

The starting point to solve the linear system in (6.1) is to calculate the eigenvalues of the equivalent matrix $\mathbf{A}$,

$$\mathbf{A} \equiv \begin{pmatrix} 5 & -2 \\ 7 & -4 \end{pmatrix}.$$

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00010-3

To derive the eigenvalues of the matrix $\mathbf{A}$ above, we calculate the determinant $|\mathbf{A} - \lambda\mathbf{I}|$. This results in the following quadratic equation to solve for λ:

$$|\mathbf{A} - \lambda\mathbf{I}| = (5-\lambda)(4-\lambda) + 14 = \lambda^2 - \lambda - 6 = (\lambda - 3)(\lambda + 2). \tag{6.2}$$

Therefore the two solutions to the quadratic equation in (6.2), and hence the eigenvalues of $\mathbf{A}$, are $\lambda_1 = 3$ and $\lambda_2 = -2$. The eigenvector associated with the two eigenvalues above can easily be shown, subject up to a constant, to be

$$\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \boldsymbol{v}_2 = \begin{pmatrix} 2 \\ 7 \end{pmatrix}.$$

Given the eigenvalues and eigenvectors derived above, it is possible to state the general solution to (6.1) as

$$\boldsymbol{x}(t) = c_1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} e^{3t} + c_2 \begin{pmatrix} 2 \\ 7 \end{pmatrix} e^{-2t}, \tag{6.3}$$

where c_1 and c_2 are constants to be determined, usually by initial conditions at time $t = 0$. The expression in (6.3) can be written componentwise as

$$\begin{aligned} x_1(t) &= c_1 e^{3t} + 2c_2 e^{-2t}, \\ x_2(t) &= c_1 c^{3t} + 7c_2 e^{-2t}. \end{aligned}$$

If we consider the initial conditions at $t = 0$ as $\boldsymbol{x}(0) = \begin{pmatrix} 0 \\ -5 \end{pmatrix}$, then the resulting simultaneous equations for c_1 and c_2 are

$$\left.\begin{aligned} 0 &= c_1 + 2c_2 \\ -5 &= c_1 + 7c_2 \end{aligned}\right\} \Rightarrow c_1 = 2, \quad c_2 = -1.$$

Therefore, given all this information, the solution to (6.1), subject to the initial conditions, is

$$\begin{aligned} x_1 &= 2e^{3t} - 2e^{-2t}, \\ x_2 &= 2e^{3t} - 7e^{-2t}. \end{aligned}$$

In many practical applications we may wish to have the solution to the linear equation take on a form/value of our choosing. Effectively we wish to ***control*** the solution so that we obtain our preferred outcome. Therefore, we need to decide in advance what we want our solution to look like and attempt to *interfere* to do so.

As in the calculus of variation chapter we shall introduce different example problems that we shall solve using the various techniques introduced throughout this chapter. The first example is the classic **predator/prey** model. The predator/prey model is also often referred to as a **population dynamics** problem.

Example 1: Predator-prey population dynamics

Let $x_1(t)$ be the total number of rabbits at time t, and let $x_2(t)$ be the total number of foxes at time t. It should be noted here that we are approximating integer values, i.e., total numbers of rabbits and foxes, by varying real numbers. It is assumed that the prey (rabbits) will grow in population if there are no predators (foxes) to reduce their numbers, and that the predators will die out if there are no prey. A simple system of ordinary differential equations that could describe this problem in general can be defined as

$$\begin{aligned} \dot{x}_1 &= a_1 x_1 - a_2 x_2, \\ \dot{x}_2 &= a_3 x_1 - a_4 x_2, \end{aligned} \tag{6.4}$$

where it is assumed that $a_i \geq 0$ for $i = 1, 2, 3, 4$. As has been shown in the first part of this chapter, it is possible to solve (6.4) using the eigenvalue-eigenvector technique.

It may be apparent that we need to interfere in this model to reduce the number of rabbits. It could be that rabbits are killed off at a rate u. Therefore, the linear system in (6.4) now becomes

$$\begin{aligned} \dot{x}_1 &= a_1 x_1 - a_2 x_2 - u, \\ \dot{x}_2 &= a_3 x_1 - a_4 x_2, \end{aligned} \tag{6.5}$$

where the control, u, is at our disposal to choose.

The system of equations in (6.5) is an example of a control system. Given this approximate real-world example, we now pose the following five questions that will be answered throughout this chapter.

1. For a given control $u = u(t)$, how does the population behave? Can we find x_1 and x_2 in terms of the control u? What this question is trying to ask is can we find the ***response*** of the system to the control u?
2. Can we obtain any level of population for the foxes and the rabbits via a judicious choice for the control u? i.e., is the system ***controllable***?
3. Suppose that we are in the situation where we can only count the total number of animals, $x_1 + x_2$, then given this total number of animals, is it possible to reconstruct the original variables x_1 and x_2? This question amounts to: is the system of ordinary differential equations ***observable***?
4. In the absence of a control to affect the populations, will the system reach some form of *equilibrium* state as $t \to \infty$? This property is asking: is the system ***stable***?
5. Is it possible to devise an automatic control strategy to maintain an equilibrium as a function of measured information? This question is equivalent to: can we derive a ***feedback***?

Example 2: Particle motion

A particle distance, s, from some fixed point has motion that may be modeled by the second-order ordinary differential equation

$$\ddot{s} = u, \tag{6.6}$$

where u is proportional to some applied force and is assumed to be under our control, and $\ddot{s} \equiv \frac{d^2 s}{dt^2}$. It is possible to write (6.6) as liner system of ordinary differential equations by letting $x_1 = s$ and $x_2 = \dot{s}$.

Differentiating x_1 and x_2 with respect to time, we have $\dot{x}_1 = \dot{s} = x_2$ and $\dot{x}_2 = \ddot{s} = u$. This results in the matrix equation

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u. \tag{6.7}$$

The matrix equation in (6.7) is of the form $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}u$.

Example 3: Load suspended by a cable

A load suspended by a cable on a horizontal track may be modeled by the equation

$$\ddot{s} = u, \quad \ddot{\theta} + \theta = u, \tag{6.8}$$

where s represents distance again, but for this example the distance is that along the track, and θ is the inclination of the cable to the vertical. The acceleration of the crane is denoted by u and may be controlled. We now need to write (6.8) as a linear system of ordinary equations. To do this we introduce

$$x_1 = s, \quad x_2 = \dot{s}, \quad x_3 = \theta, \quad x_4 = \dot{\theta}.$$

Differentiating the new variable with respect to t results in

$$\dot{x}_1 = \dot{s} = x_2, \quad \dot{x}_2 = \ddot{s} = u, \quad \dot{x}_3 = \dot{\theta} = x_4, \quad \dot{x}_4 = \ddot{\theta} = -\theta + u = -x_3 + u.$$

Writing the expression above in the form $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}u$ results in

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix}, \quad \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}. \tag{6.9}$$

In the three examples presented above, it is possible that we can *control* u so that $\boldsymbol{x}$ behaves in a manner that we deem to be desirable. For all three examples above, we require initial conditions to ensure that the solutions to the linear system of ordinary differential equations are unique. Thus, we require $\boldsymbol{x}(t_0)$ to be provided.

So far, in the three examples presented, we have only had a single control. It is more likely that there could be several controls.

Example 4: Multiple controls

Consider the linear system of ordinary differential equations

$$\begin{aligned} \dot{x}_1 &= x_1 + x_3 - u_1 + 2u_2, \\ \dot{x}_2 &= x_3 + u_1, \\ \dot{x}_3 &= x_2 - 3u_2, \end{aligned} \tag{6.10}$$

where we now have the two controls u_1 and u_2. Unlike in the first three example, $\mathbf{B}$ is now a matrix rather than a vector. Therefore, we can write (6.10) as the matrix equation

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix}, \quad \boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} -1 & 2 \\ 1 & 0 \\ 0 & -3 \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}.$$

The matrix equation has changed slightly from its form presented earlier. The expression above for the system of linear ordinary differential equations in (6.10) is of the form $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$.

Recalling the predator-prey model, we stated that it may only be possible to observe the total number of animals. For this multiple control problem we may only be able to observe certain combinations of the x_i, $i = 1, 2, 3$. The observed states are referred to as *outputs*. However, the observed states are denoted by y_j, which is the notation for observations in data assimilation. For this example we are going to introduce the observed states as

$$\begin{aligned} y_1 &= x_1 + 2x_2, \\ y_2 &= x_2 - x_3. \end{aligned}$$

As with the rest of the linear system, the outputs can be written in matrix form as $\boldsymbol{y} = \mathbf{C}\boldsymbol{x}$, where for the observations above

$$\mathbf{C} = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & -1 \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}.$$

General form of a control problem

The most general matrix formulation of the control problem is

$$\begin{aligned} \dot{\boldsymbol{x}} &= \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \\ \boldsymbol{y} &= \mathbf{C}\boldsymbol{x}, \end{aligned} \tag{6.11}$$

where

$$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{pmatrix}, \quad \boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix},$$

and $\mathbf{A}$ is $n \times n$, $\mathbf{B}$ is $n \times m$ and $\mathbf{C}$ is $p \times n$. The variables $x_1, x_2, \ldots, x_n$ are the ***state variables***, $u_1, u_2, \ldots, u_m$ are the ***control variables***, and $y_1, y_2, \ldots, y_p$ are the ***output variables***.

6.2 The Uncontrolled Problem

The starting point for control theory is to consider the uncontrolled initial value problem $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$ for $t > t_0$, given the initial conditions $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, where $\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$. The matrix $\mathbf{A}$ is $n \times n$ matrix whose

entries are continuous functions of t for $t > t_0$. Under the circumstances (assumptions) just described, there exists a unique solution $\boldsymbol{x} = \boldsymbol{x}(t)$.

6.2.1 Fundamental Solutions

If we consider the n separate initial value problems defined by

$$\left.\begin{array}{l} \dot{\boldsymbol{\phi}}_i = \mathbf{A}\boldsymbol{\phi}_i \qquad (t > t_0) \\ \boldsymbol{\phi}_i(t_0) = \boldsymbol{e}_i \end{array}\right\} \quad i = 1, 2, \ldots, n, \tag{6.12}$$

where the vectors $\boldsymbol{e}_i$ are the unit vectors such that

$$\boldsymbol{e}_i^T = \begin{pmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{pmatrix},$$

and the 1 is in the ith position of the vector, then it is clear that any linear combination of the $\boldsymbol{\phi}_i$ will also solve the differential system.

If we now define a vector $\boldsymbol{z}$ to be

$$\boldsymbol{z} = c_1\boldsymbol{\phi}_1 + c_2\boldsymbol{\phi}_2 + \cdots + c_n\boldsymbol{\phi}_n, \tag{6.13}$$

then it is clear that $\dot{\boldsymbol{z}} = \mathbf{A}\boldsymbol{z}$. If we now consider the initial conditions for $\boldsymbol{z}$, then we have

$$\boldsymbol{z}(t_0) = c_1\boldsymbol{e}_1 + c_2\boldsymbol{e}_2 + \cdots + c_n\boldsymbol{e}_n = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} = \boldsymbol{x}_0. \tag{6.14}$$

If we choose the constants c_i to be the ith entry in the initial condition vector $\boldsymbol{x}_0$, then it follows that $\boldsymbol{z} = \boldsymbol{x}$. Therefore, we have

$$\begin{aligned} \boldsymbol{x}(t) &= c_1\boldsymbol{\phi}_1(t) + c_2\boldsymbol{\phi}_2(t) + \cdots + c_n\boldsymbol{\phi}_n(t), \\ &= \begin{bmatrix} \boldsymbol{\phi}_1(t) & \boldsymbol{\phi}_2(t) & \cdots & \boldsymbol{\phi}_n(t) \end{bmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}, \\ &= \begin{bmatrix} \boldsymbol{\phi}_1(t) & \boldsymbol{\phi}_2(t) & \cdots & \boldsymbol{\phi}_n(t) \end{bmatrix} \boldsymbol{x}_0. \end{aligned} \tag{6.15}$$

The matrix defined by (6.15) depends upon both t and the initial time t_0 and is referred to in the control theory literature as the ***state transition matrix***. This matrix is usually denoted as $\boldsymbol{\Phi}(t, t_0)$.

Example 6.1. *Find the state transition matrix of the system*

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}. \tag{6.16}$$

If we take each row of (6.16) after following the matrix-vector multiplication rule, then we obtain

$$\begin{array}{l} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_2 \end{array} \quad \Rightarrow \quad \begin{array}{l} x_2 = Ae^{-t}, \\ x_1 = -Ae^{-t} + B. \end{array}$$

We now use the condition that the fundamental solutions, $\boldsymbol{\phi}_i$s, at the initial time, t_0 are equal to the unit vectors which provides the conditions to set the values for the constants of integration for each of the fundamental solutions.

$$\begin{aligned}\boldsymbol{\phi}_1(t_0) &= \begin{pmatrix} -Ae^{-t_0} + B \\ Ae^{-t_0} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \\ &\Rightarrow A = 0, B = 1, \\ &= \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \end{aligned} \tag{6.17}$$

$$\begin{aligned}\boldsymbol{\phi}_2(t_0) &= \begin{pmatrix} -Ae^{-t_0} + B \\ Ae^{-t_0} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\ &\Rightarrow A = e^{t_0}, B = 1, \\ &= \begin{pmatrix} -e^{t_0 - t} + 1, \\ e^{t_0 - t} \end{pmatrix}. \end{aligned} \tag{6.18}$$

Combining (6.17) and (6.18), we can form the state transition matrix for (6.16) as

$$\begin{aligned}\boldsymbol{\Phi}(t, t_0) &= \begin{bmatrix} \boldsymbol{\phi}_1(t) & \boldsymbol{\phi}_2(t) \end{bmatrix}, \\ &= \begin{pmatrix} 1 & 1 - e^{t_0 - t} \\ 0 & e^{t_0 - t} \end{pmatrix}. \end{aligned} \tag{6.19}$$

6.2.2 Properties of the State Transition Matrix

The main property of the state transition matrix has already been presented, which is that the solution of the first-order system of initial value problems,

$$\boldsymbol{x}(t) = \mathbf{A}\boldsymbol{x} \quad (t > t_0), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0,$$

can be written as $\boldsymbol{x}(t) = \boldsymbol{\Phi}(t, t_0)\boldsymbol{x}_0$. It has also been shown that the state transition matrix can be found in terms of fundamental solutions of the differential equations, $\boldsymbol{\phi}_i(t)$. We now consider four more very important properties of the state transition matrix:

1. $\frac{d\boldsymbol{\Phi}}{dt} = \mathbf{A}\boldsymbol{\Phi}$. An important feature to note here is that we are equating two matrices.
   Proof: This proof is achieved through writing $\boldsymbol{\Phi}$ in terms of the fundamental solutions. We start with

$$\begin{aligned}\frac{d\boldsymbol{\Phi}}{dt} &= \frac{d}{dt}\begin{bmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \dots & \boldsymbol{\phi}_n \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{\phi}}_1 & \dot{\boldsymbol{\phi}}_2 & \dots & \dot{\boldsymbol{\phi}}_n \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A}\boldsymbol{\phi}_1 & \mathbf{A}\boldsymbol{\phi}_2 & \dots & \mathbf{A}\boldsymbol{\phi}_n \end{bmatrix} = \mathbf{A}\begin{bmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \dots & \boldsymbol{\phi}_n \end{bmatrix}, \\ &= \mathbf{A}\boldsymbol{\Phi}, \end{aligned}$$

   as required.

2. $\mathbf{\Phi}(t_0, t_0) = \mathbf{I}$, the $n \times n$ identity matrix.
Proof: The proof for this property has already been show implicitly in the introduction of the state transition matrix. The proof uses the property that $\boldsymbol{\phi}_i(t_0) = \boldsymbol{e}_i$. Therefore,

$$\begin{aligned}\mathbf{\Phi}(t_0, t_0) &= \begin{bmatrix} \boldsymbol{\phi}_1(t_0) & \boldsymbol{\phi}_2(t_0) & \dots & \boldsymbol{\phi}_n(t_0) \end{bmatrix} = \begin{bmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \dots & \boldsymbol{e}_n \end{bmatrix}, \\ &= \mathbf{I}.\end{aligned}$$

3. $\mathbf{\Phi}(t_2, t_0) = \mathbf{\Phi}(t_2, t_1)\,\mathbf{\Phi}(t_1, t_0)$
Proof: From the definition of the state transition matrix, we know that

$$\boldsymbol{x}(t_2) = \mathbf{\Phi}(t_2, t_0)\boldsymbol{x}(t_0)$$

and

$$\boldsymbol{x}(t_2) = \mathbf{\Phi}(t_2, t_1)\boldsymbol{x}(t_1) = \mathbf{\Phi}(t_2, t_1)\,\mathbf{\Phi}(t_1, t_0)\boldsymbol{x}(t_0).$$

Eliminating $\boldsymbol{x}(t_2)$ from the two expressions above by subtracting the second equation from the first equation results in

$$(\mathbf{\Phi}(t_2, t_0) - \mathbf{\Phi}(t_2, t_1)\,\mathbf{\Phi}(t_1, t_0))\boldsymbol{x}(t_0) = \mathbf{0},$$

which must be true for sets of initial conditions not just $\boldsymbol{x}(t_0) = \mathbf{0}$. Therefore, the only way the expression above can hold true for all initial conditions is if $\mathbf{\Phi}(t_2, t_0) = \mathbf{\Phi}(t_2, t_1)\,\mathbf{\Phi}(t_1, t_0)$ which proves our required property.

4. The final property of the state transition matrix involves its inverse; $\mathbf{\Phi}(t, t_0)^{-1} = \mathbf{\Phi}(t_0, t)$.
Proof: The proof of this property involves using the two previous properties:

$$\mathbf{\Phi}(t_0, t)\,\mathbf{\Phi}(t, t_0) = \mathbf{\Phi}(t_0, t_0) = \mathbf{I}.$$

Exercise 6.2. *Suppose that* $\boldsymbol{x}(t) = \mathbf{\Phi}(t, t_0)\boldsymbol{x}_0$ *is the solution to the initial value problem*

$$\begin{aligned}\dot{\boldsymbol{x}}(t) &= \mathbf{A}\boldsymbol{x}(t) \quad (t > t_0), \\ \boldsymbol{x}(t_0) &= \boldsymbol{x}_0,\end{aligned}$$

where $\mathbf{\Phi}(t, t_0)$ *is the state transition matrix. Find* $\mathbf{\Phi}(t, t_0)$ *for each of the following choices of* $\mathbf{A}$*:*

$$(a) \quad \mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & -2 \end{pmatrix}, \quad (b) \quad \mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}.$$

6.2.3 Time-Invariant Case

We now consider the solution solver for the situation where the matrix $\mathbf{A}$ is a constant with respect to time. Under this situation it is possible to write the state transition matrix in a different way. We start with the following definition.

Definition 6.3. If $\mathbf{A}$ is a constant matrix then the exponential matrix, denoted either as $\exp\{\mathbf{A}\}$ or $e^{\mathbf{A}}$, is defined by

$$\exp\{\mathbf{A}\} = \mathbf{I} + \mathbf{A} + \frac{1}{2}\mathbf{A}^2 + \frac{1}{3!}\mathbf{A}^3 + \cdots + \frac{1}{n!}\mathbf{A}^n + \cdots, \tag{6.20}$$

$$= \sum_{i=1}^{\infty} \frac{1}{i!}\mathbf{A}^i. \tag{6.21}$$

Given Definition 6.3, it is now possible to state and prove the following theorem:

Theorem 6.4. *The matrix* $\exp\{(t-t_0)\boldsymbol{A}\}$ *satisfies the matrix initial value problem:*

$$\frac{d}{dt}\boldsymbol{X} = \boldsymbol{A}\boldsymbol{X},$$
$$\boldsymbol{X}(t_0) = \boldsymbol{I}$$

and is therefore the state transition matrix $\boldsymbol{\Phi}(t, t_0)$.

Proof. The proof of Theorem 6.4 starts from considering (6.20) in Definition 6.3:

$$\exp\{(t-t_0)\} = \mathbf{I} + (t-t_0)\mathbf{A} + \frac{(t-t_0)^2}{2}\mathbf{A}^2 + \frac{(t-t_0)^3}{3!}\mathbf{A}^3 + \cdots + \frac{(t-t_0)^n}{n!}\mathbf{A}^n + \cdots. \tag{6.22}$$

The next step is to differentiate each of the terms in (6.22) with respect to t:

$$\begin{aligned}\frac{d}{dt}\exp\{(t-t_0)\} &= \mathbf{A} + \frac{2(t-t_0)^2}{2}\mathbf{A}^2 + \frac{3(t-t_0)^2}{3}\mathbf{A}^3 + \cdots + \frac{n(t-t_0)^{n-1}}{n!}\mathbf{A}^{n-1} + \cdots, \\ &= \mathbf{A}\left(\mathbf{I} + (t-t_0)\mathbf{A} + \frac{(t-t_0)^2}{2}\mathbf{A}^2 + \cdots + \frac{(t-t_0)^{n-1}}{(n-1)!}\mathbf{A}^{n-1} + \cdots\right), \\ &= \mathbf{A}\exp\{(t-t_0)\},\end{aligned}$$

which is in the form stated in Theorem 6.4. To complete the proof, we require the property that $\exp\{(t_0-t_0)\} = \mathbf{I}$, which can clearly be seen by evaluating (6.22) at $t = t_0$ where all the terms in the summation become zero except the first term, which is the identity, and hence the theorem is proven.

Theorem 6.4 is only true for constant $\mathbf{A}$ because if we consider a time variant matrix, $\mathbf{A} = \mathbf{A}(t)$, then differentiating $\mathbf{A}^2$ has to be done by the product rule, and so

$$\begin{aligned}\frac{d}{dt}(\mathbf{A})^2 = \frac{d}{dt}(\mathbf{A}\mathbf{A}) &= \left(\frac{d}{dt}\mathbf{A}\right)\mathbf{A} + \mathbf{A}\left(\frac{d}{dt}\mathbf{A}\right), \\ &\neq 2\mathbf{A}\left(\frac{d}{dt}\mathbf{A}\right), \quad \text{in general.}\end{aligned}$$

As an example of the theory just presented, we consider Example 2: the particle in motion but without the control. We have shown that the linear systems of ordinary differential equations can be written with a constant $\mathbf{A}$ matrix as

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

Using Definition 6.3, we have that

$$\mathbf{\Phi}(t, t_0) = e^{(t-t_0)\mathbf{A}} = \mathbf{I} + (t - t_0)\mathbf{A} + \frac{(t-t_0)^2}{2}\mathbf{A}^2 + \cdots$$

However, upon multiplying $\mathbf{AA}$ we see that this results in $\mathbf{A}^2 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, which terminates all the higher-order powers in the series and as such we only require the first two terms. Therefore, the state transition matrix for the uncontrolled version of the object in motion is

$$e^{(t-t_0)\mathbf{A}} = \mathbf{I} + (t - t_0)\mathbf{A} = \begin{pmatrix} 1 & t - t_0 \\ 0 & 1 \end{pmatrix}. \tag{6.23}$$

Exercise 6.5. *Verify that the state transition matrix found earlier for the system based upon the matrix* $\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}$ *is equivalent to* $\exp\{(t - t_0)\mathbf{A}\}$.

6.2.4 Properties of Exponential Matrices

We now present four important properties of exponential matrices and show their short proofs.

1. For the two matrices $\mathbf{A}$ and $\mathbf{B}$ that commute, then $e^{\mathbf{A}}e^{\mathbf{B}} = e^{\mathbf{A}+\mathbf{B}}$.
   Proof:

$$\begin{aligned} e^{\mathbf{A}}e^{\mathbf{B}} &= \left(\mathbf{I} + \mathbf{A} + \frac{1}{2}\mathbf{A}^2 + \frac{1}{3!}\mathbf{A}^3 + \cdots\right)\left(\mathbf{I} + \mathbf{B} + \frac{1}{2}\mathbf{B}^2 + \frac{1}{3!}\mathbf{B}^3 + \cdots\right), \\ &= \mathbf{I} + (\mathbf{A} + \mathbf{B}) + \frac{1}{2}\left(\mathbf{A}^2 + 2\mathbf{AB} + \mathbf{B}^2\right) + \frac{1}{3!}\left(\mathbf{A}^3 + 3\mathbf{A}^2\mathbf{B} + 3\mathbf{AB}^2 + \mathbf{B}^3\right) + \cdots, \\ &= \mathbf{I}(\mathbf{A} + \mathbf{B}) + \frac{1}{2}(\mathbf{A} + \mathbf{B})^2 + \frac{1}{3!}(\mathbf{A} + \mathbf{B})^3 + \cdots, \\ &= e^{\mathbf{A}+\mathbf{B}}. \end{aligned}$$

   Note: We have only been able to obtain this result due to the commutability property. In general, the factorizations above would not come about as it is not true for all matrices that $(\mathbf{A} + \mathbf{B})^2 = (\mathbf{A} + \mathbf{B})(\mathbf{A} + \mathbf{B}) = \mathbf{A}^2 + \mathbf{AB} + \mathbf{BA} + \mathbf{B}^2 \neq \mathbf{A}^2 + 2\mathbf{AB} + \mathbf{B}^2$.
2. The exponential matrix $e^{\mathbf{A}}$ is invertible for any matrix $\mathbf{A}$.
   Proof: The first feature to note here is that $\mathbf{A}$ and $-\mathbf{A}$ commute, $\mathbf{A}(-\mathbf{A}) = -\mathbf{A}^2 = -\mathbf{A}(\mathbf{A})$, which means that we can use the first property just proven. Therefore,

$$e^{\mathbf{A}}e^{-\mathbf{A}} = e^{\mathbf{A}-\mathbf{A}} = e^{\mathbf{0}} = \mathbf{I},$$

   where $\mathbf{0}$ is a $n \times n$ matrix of zeros. As a by-product of this proof we have that $\left(e^{\mathbf{A}}\right)^{-1} = e^{-\mathbf{A}}$.
3. Given the result above the inverse, we can now state that $\left(e^{(t-t_0)\mathbf{A}}\right)^{-1} = e^{(t_0-t)\mathbf{A}}$. This statement has already been proven through the properties of the state transition matrices earlier.
4. If we have a diagonal matrix $\mathbf{D}$, such that $\mathbf{D} = diag\{d_1, d_2, \ldots, d_n\}$, then

$$e^{\mathbf{D}} = \mathbf{I} + \mathbf{D} + \frac{1}{2}\mathbf{D}^2 + \frac{1}{3!}\mathbf{D}^3 + \cdots. \tag{6.24}$$

Proof:

$$
\begin{aligned}
e^{\mathbf{D}} &= \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} + \begin{pmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{pmatrix} + \frac{1}{2}\begin{pmatrix} d_1^2 & & & \\ & d_2^2 & & \\ & & \ddots & \\ & & & d_n^2 \end{pmatrix} + \cdots, \\
&= \begin{pmatrix} 1 + d_1 + \frac{1}{2}d_1^2 + \cdots & & & \\ & 1 + d_2 + \frac{1}{2}d_2^2 + \cdots & & \\ & & \ddots & \\ & & & 1 + d_n + \frac{1}{2}d_n^2 + \cdots \end{pmatrix}, \\
&= \begin{pmatrix} e^{d_1} & & & \\ & e^{d_2} & & \\ & & \ddots & \\ & & & e^{d_n} \end{pmatrix}.
\end{aligned}
$$

So far, we have introduced two different techniques for finding the state transition matrix through the exponential matrix approach:

1. solving the differential equations in general; and
2. through good fortune that the second-order term in the exponential matrix expansion is zero, i.e., $\mathbf{A}^2 = \mathbf{0}$.

A more general approach to find the exponential matrix is through eigenvalues and eigenvectors.

6.2.5 Eigenvalues/Vectors Approach for Finding the State Transition Matrix

We have already shown that a solution to the system of ordinary differential equations can be expressed as

$$\dot{\boldsymbol{x}} = e^{(t-t_0)\mathbf{A}}\boldsymbol{x}(t_0).$$

We now consider a new approach for finding the state transition matrix involving eigenvalues and eigenvectors. We will denote the eigenvalues and eigenvectors of $\mathbf{A}$ by λ_i and $\boldsymbol{v}_i$, respectively. We now assume that the general solution of $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$ can be written as

$$\boldsymbol{x}(t) = \mathbf{V}\mathbf{D}(t)\boldsymbol{c}, \tag{6.25}$$

where

- $\mathbf{V}$ is a non-singular $n \times n$ time independent matrix;
- $\mathbf{D}(0) = \mathbf{I}$; and
- $\boldsymbol{c}$ contains the arbitrary constants.

Given the three assumptions above, we formulate that the fundamental solutions can be defined as

$$\begin{aligned}\boldsymbol{\phi}_i(t) &= \mathbf{V}\mathbf{D}(t)\boldsymbol{c}_i, \quad \text{where} \quad \boldsymbol{c}_i \quad \text{s.t.} \quad \boldsymbol{\phi}_i(0) = \boldsymbol{e}_i,\\ \boldsymbol{\phi}_i(0) &= \mathbf{V}\mathbf{D}(0)\boldsymbol{c}_i = \mathbf{V}\boldsymbol{c}_i \Rightarrow \boldsymbol{c}_i = \mathbf{V}^{-1}\boldsymbol{e}_i.\end{aligned}$$

Therefore, we can write the state transition matrix as

$$\begin{aligned}\boldsymbol{\Phi}(t,0) = e^{t\mathbf{A}} &= \left[\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_n\right],\\ &= \mathbf{V}\mathbf{D}(t)\mathbf{V}^{-1}\underbrace{[\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_n]}_{\mathbf{I}},\\ \Rightarrow e^{t\mathbf{A}} &= \mathbf{V}\mathbf{D}(t)\mathbf{V}^{-1},\\ e^{(t-t_0)\mathbf{A}} &= \mathbf{V}\mathbf{D}(t-t_0)\mathbf{V}^{-1}.\end{aligned} \tag{6.26}$$

We now consider different circumstances that could occur when finding the eigenvalues of the $\mathbf{A}$ matrix.

Case 1: The eigenvectors of A are linearly independent

An implication of the eigenvectors being linearly independent is that $\mathbf{A}$ is diagonalizable. However, it is possible that the eigenvalues could all be real numbers or complex numbers. We take each situation in turn.

Case 1a: Eigenvalues $\lambda_i \in \mathbb{R}$, $i = 1, 2, \ldots, n$

The starting point is to assume that a solution to $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$ is given by $\boldsymbol{x}(t) = e^{\lambda_i t}\boldsymbol{v}_i$, which is verified by

$$\dot{\boldsymbol{x}}(t) = \lambda_i e^{\lambda_i t}\boldsymbol{v}_i = e^{\lambda_i t}\mathbf{A}\boldsymbol{v}_i = \mathbf{A}\boldsymbol{x}(t).$$

Therefore, it is possible to state the general solution as

$$\begin{aligned}\boldsymbol{x}(t) &= \boldsymbol{c}_1 e^{\lambda_1 t}\boldsymbol{v}_1 + \boldsymbol{c}_2 e^{\lambda_2 t}\boldsymbol{v}_2 + \cdots + \boldsymbol{c}_n e^{\lambda_n t}\boldsymbol{v}_n,\\ &= \begin{pmatrix} e^{\lambda_1 t}\boldsymbol{v}_1 & e^{\lambda_2 t}\boldsymbol{v}_2 & \cdots & e^{\lambda_n t}\boldsymbol{v}_n \end{pmatrix}\begin{pmatrix} \boldsymbol{c}_1\\ \boldsymbol{c}_2\\ \vdots\\ \boldsymbol{c}_n \end{pmatrix},\\ &= \begin{pmatrix} \boldsymbol{v}_1 & \boldsymbol{v}_2 & \cdots & \boldsymbol{v}_n \end{pmatrix}\begin{pmatrix} e^{\lambda_1 t} & & & \\ & e^{\lambda_2 t} & & \\ & & \ddots & \\ & & & e^{\lambda_n t} \end{pmatrix}\begin{pmatrix} \boldsymbol{c}_1\\ \boldsymbol{c}_2\\ \vdots\\ \boldsymbol{c}_n \end{pmatrix},\\ &= \mathbf{V}\mathbf{D}(t)\boldsymbol{c}.\end{aligned}$$

As a result of the eigenvectors being linearly independent, the matrix containing them, $\mathbf{V}$, is non-singular and hence invertible. Therefore, we can state that the state transition matrix for this situation is

$$e^{t\mathbf{A}} = \mathbf{V}\mathbf{D}(t)V^{-1}.$$

Case 1b: Some eigenvalues are complex

We now move to the situation where, say, the first two eigenvalues, λ_1 and λ_2, are complex numbers, i.e., $\lambda_1, \lambda_2 \in \mathbb{C}$. The same arguments that were used in Case 1a can be applied in this situation, but we have the situation where some of the columns of $\mathbf{V}$ are complex and some of the elements in $\mathbf{D}$ are complex. We wish to rearrange them so that $\mathbf{V}$ and $\mathbf{D}$ are expressed in terms of real quantities.

Without loss of generality, we assume that it is only the first two eigenvalues that are complex numbers and the remaining $n-2$ eigenvalues are real numbers, i.e., $\lambda_{1,2} \in \mathbb{C}$ and $\lambda_3, \ldots, \lambda_n \in \mathbb{R}$, where $\lambda_{1,2} = \alpha \pm i\beta$, $\beta \neq 0$ and $\boldsymbol{v}_{1,2} = \mathbf{p} \pm i\mathbf{q}$ where $\alpha, \beta \in \mathbb{R}$ and $\mathbf{p}, \mathbf{q} \in \mathbb{R}^n$.

We know that $e^{\lambda_j t}\mathbf{v}_j$ satisfies the system of linear ordinary differential equations $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$. Therefore, we can rewrite the first two solutions as

$$\begin{aligned} e^{\lambda_{1,2} t}\boldsymbol{v}_{1,2} &= e^{\alpha t}\left(\cos\beta t \pm i \sin\beta t\right)(\boldsymbol{p} + i\boldsymbol{q}), \\ &= e^{\alpha t}\left(\boldsymbol{p}\cos\beta t - \boldsymbol{q}\sin\beta t\right) \pm i e^{\alpha t}\left(\boldsymbol{p}\sin\beta t + \boldsymbol{q}\cos\beta t\right). \end{aligned} \tag{6.27}$$

We require both the real and complex component to be solutions of the linear system of ordinary differential equations, therefore we form the general solution as

$$\begin{aligned} \boldsymbol{x}(t) &= c_1 e^{\alpha t}\left(\boldsymbol{p}\cos\beta t - \boldsymbol{q}\sin\beta t\right) + c_2 e^{\alpha t}\left(\boldsymbol{p}\sin\beta t + \boldsymbol{q}\cos\beta t\right) + \sum_{j=3}^{n} c_j e^{\lambda_j t}\boldsymbol{v}_j, \\ &= e^{\alpha t}\begin{pmatrix} \boldsymbol{p} & \boldsymbol{q} \end{pmatrix}\begin{pmatrix} \cos\beta t & \sin\beta t \\ -\sin\beta t & \cos\beta t \end{pmatrix}\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \\ &\quad + \begin{pmatrix} \boldsymbol{v}_3 & \ldots & \boldsymbol{v}_n \end{pmatrix}\begin{pmatrix} e^{\lambda_3 t} & & \\ & \ddots & \\ & & e^{\lambda_n} \end{pmatrix}\begin{pmatrix} c_3 \\ \vdots \\ c_n \end{pmatrix}, \\ &= \begin{pmatrix} \boldsymbol{p} & \boldsymbol{q} & \boldsymbol{v}_3 & \ldots & \boldsymbol{v}_n \end{pmatrix}\begin{pmatrix} e^{\alpha t}\begin{pmatrix} \cos\beta t & \sin\beta t \\ -\sin\beta t & \cos\beta t \end{pmatrix} & & & \\ & e^{\lambda_3 t} & & \\ & & \ddots & \\ & & & e^{\lambda_n t} \end{pmatrix}\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{pmatrix}. \end{aligned} \tag{6.28}$$

The solution in (6.28) is still in the diagonal form as presented for the eigenvalue approach, but now we have a block diagonal matrix and that $\mathbf{V}$ is still invertible.

We now consider a simple example to illustrate that the technique above is applicable.

Example 6.6. *Find the state transition matrix for the uncontrolled system with*

$$\boldsymbol{A} = \begin{pmatrix} 0 & 1 \\ -\nu^2 & 0 \end{pmatrix}.$$

The eigenvalues for the matrix in Example 6.6 can easily be shown to be $\lambda_1 = i\nu$ and $\lambda_2 = -i\nu$ that have corresponding eigenvectors $\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ i\nu \end{pmatrix}$ and $\boldsymbol{v}_2 = \begin{pmatrix} 1 \\ -i\nu \end{pmatrix}$. Employing the notation used

to derive (6.28), we have

$$\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} + i \begin{pmatrix} 0 \\ \nu \end{pmatrix}, \quad \lambda_1 = 0 + i\nu,$$

which gives us

$$\mathbf{V} = \begin{pmatrix} \boldsymbol{p} & \boldsymbol{q} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & \nu \end{pmatrix} \Rightarrow \mathbf{V}^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\nu} \end{pmatrix}.$$

Therefore, the state transition matrix for Example 6.6 is

$$\begin{aligned} e^{t\mathbf{A}} &= e^{\alpha t} \begin{pmatrix} 1 & 0 \\ 0 & \nu \end{pmatrix} \begin{pmatrix} \cos \nu t & \sin \nu t \\ -\sin \nu t & \cos \nu t \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\nu} \end{pmatrix}, \\ &= \begin{pmatrix} \cos \nu t & \frac{1}{\nu} \sin \nu t \\ -\nu \sin \nu t & \cos \nu t \end{pmatrix}. \end{aligned}$$

Case 2: The matrix A has repeated eigenvalues and also has less than n linearly independent eigenvectors

There are two cases that we consider here. The first is that there are $n-1$ linearly independent eigenvectors and the second is when we have less than $n-1$ linearly independent eigenvectors.

Case 2a: There are $n-1$ linearly independent eigenvectors

The major implication for a repeated eigenvalue with only one eigenvector is that we cannot use the approach that we have shown for the first set of cases. We therefore need a different technique to find $\mathbf{V}$.

Without loss of generality, we will assume that it is $\lambda_1 = \lambda_2$ and that the eigenvectors associated with the remaining distinct eigenvalues along with the eigenvector for the first eigenvalue form the linearly independent set of eigenvectors $\{\boldsymbol{v}_1, \boldsymbol{v}_3, \ldots, \boldsymbol{v}_n\}$.

To find the extra solution, we borrow a technique used to solve differential equations, which is what control theory is based upon. We assume that there is a solution to $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$ that is of the form

$$\boldsymbol{x} = e^{\lambda_1 t} (\boldsymbol{w} + t\boldsymbol{v}_1), \tag{6.29}$$

where $\boldsymbol{w}$ is to be found. Therefore, differentiating (6.29), we have

$$\begin{aligned} \dot{\boldsymbol{x}} &= \lambda_1 e^{\lambda_1 t} (\boldsymbol{w} + t\boldsymbol{v}_1) + e^{\lambda_1 t} \boldsymbol{v}_1, \\ \mathbf{A}\boldsymbol{x} &= e^{\lambda_1 t} (\mathbf{A}\boldsymbol{w} + t\mathbf{A}\boldsymbol{v}_1) = e^{\lambda_1 t} (\mathbf{A}\boldsymbol{w} + t\lambda_1 \boldsymbol{v}_1). \end{aligned}$$

Equating the two expressions above yields

$$\begin{aligned} &\lambda_1 \boldsymbol{w} + \lambda_1 t\boldsymbol{v}_1 + \boldsymbol{v}_1 = \mathbf{A}\boldsymbol{w} + \lambda_1 t\boldsymbol{v}_1, \\ &\Rightarrow (\mathbf{A} - \lambda_1 \mathbf{I})\, \boldsymbol{w} = \boldsymbol{v}_1, \end{aligned} \tag{6.30}$$

where we assume that (6.30) has a solution.

Therefore, the general solution for the set of linear ordinary differential equations where there are repeated eigenvalues, but not a complete set of initial eigenvectors, is given by

$$\begin{aligned}\boldsymbol{x}(t) &= c_1 e^{\lambda_1 t}\boldsymbol{v}_1 + c_2 e^{\lambda_1 t}(\boldsymbol{w} + t\boldsymbol{v}_1) + c_3 e^{\lambda_3 t}\boldsymbol{v}_3 + \cdots + c_n e^{\lambda_n t}\boldsymbol{v}_n,\\ &= \begin{pmatrix} \boldsymbol{v}_1 & \boldsymbol{w} & \boldsymbol{v}_3 & \dots & \boldsymbol{v}_n \end{pmatrix} \begin{pmatrix} e^{\lambda_1 t}\begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} & & & \\ & e^{\lambda_3 t} & & \\ & & \ddots & \\ & & & e^{\lambda_n t} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_n \end{pmatrix}. \end{aligned} \tag{6.31}$$

An inverse to $\mathbf{V}$ does exist, but the proof involves proof by contradiction and can be found in [25].

Example 6.7. *Returning to Example 2 for this chapter, we have* $\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$. *Find the state transition matrix through the eigenvalue-eigenvector approach.*

The eigenvalues for the $\mathbf{A}$ matrix in Example 2 can easily be shown to be $\lambda_1 = \lambda_2 = 0$. The eigenvector is derived as

$$(\mathbf{A} - \lambda_1 \mathbf{I})\boldsymbol{v}_1 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\boldsymbol{v}_1 = \mathbf{0} \Rightarrow \boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

but up to a multiplication of a constant. We therefore know that the first solution is $\boldsymbol{v}_1 e^{\lambda_1 t}$. The second solution of the system in Example 2 has been proven to be of the form $(\boldsymbol{w} + t\boldsymbol{v}_1)e^{\lambda_1 t}$, where $\boldsymbol{w}$ is still to be found. Constructing the solver, $(\mathbf{A} - \lambda_1 \mathbf{I})\boldsymbol{w} = \boldsymbol{v}_1$, requires

$$\begin{aligned}\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} &= \begin{pmatrix} 1 \\ 0 \end{pmatrix},\\ \Rightarrow \quad w_2 = 1, &\qquad w_1 = c. \end{aligned}$$

Without loss of generality we can set the constant in w_1 equal to zero; this then makes $\boldsymbol{w} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Collecting all the terms to form the state transition matrix for Example 2, we have

$$\begin{aligned} e^{t\mathbf{A}} &= \mathbf{V}\mathbf{D}(t)\mathbf{V}^{-1},\\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} e^{\lambda_1 t}\begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},\\ &= \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}, \end{aligned}$$

which agrees with the solutions we found earlier.

Case 2b: Fewer still eigenvectors

We only consider the case where there is one more eigenvector missing in Case 2a. The techniques we present here are extendable to a higher number of missing of eigenvectors.

Therefore, assuming multiplicity of order 3 for the first eigenvalue. We have already shown that to find the first missing eigenvector, we solve for $\boldsymbol{w}_1$ through $(\mathbf{A} - \lambda_1 \mathbf{I})\boldsymbol{w}_1 = \boldsymbol{v}_1$. We saw that extending a

technique for solving differential equations enabled us to derive a solver to find a second eigenvector in Case 2a. It is from the differential equations theory that the third eigenvector is constructed. Without deriving the expression, it can be shown that the third eigenvector, $\boldsymbol{w}_2$, is such that $(\mathbf{A} - \lambda_1 \mathbf{I})\,\boldsymbol{w}_2 = \boldsymbol{w}_1$, which then leads to the general solution as

$$e^{t\mathbf{A}} = c_1 e^{\lambda_1 t}\boldsymbol{v}_1 + c_2 e^{\lambda_1 t}\left(\boldsymbol{w}_1 + t\boldsymbol{v}_1\right) + c_3 e^{\lambda_1 t}\left(\boldsymbol{w}_2 + t\boldsymbol{w}_1 + t^2\boldsymbol{v}_1\right) + \sum_{j=4}^{n} c_j e^{\lambda_j t}\boldsymbol{v}_j. \tag{6.32}$$

6.3 The Controlled Problem

In this section we present techniques and examples, along with exercises, to solve the controlled problem. The general matrix-vector equation for the controlled problem is defined as

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \quad (t > t_0), \tag{6.33}$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0. \tag{6.34}$$

In the previous section we presented techniques to find the complementary functions for the homogeneous problem. As mentioned in Section 6.2, the uncontrolled problem was solved by using techniques from differential equation theory. We shall look to this same theory to develop solvers for the controlled problem. We therefore introduce a particular integral, $\boldsymbol{x}_{pi}$, which can be any solution to $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$. As it has already been shown that the complementary functions satisfies the initial conditions for the uncontrolled problem, it is inferred then that the particular integral should satisfy the initial condition $\boldsymbol{x}_{pi}(t_0) = \mathbf{0}$.

Given the solution that we seek, we now require a technique to find the particular integral. The theory that we shall use is referred to as a variation of parameters. The starting point is to assume that the particular integral can be written as

$$\boldsymbol{x}_{pi}(t) = \boldsymbol{\Phi}(t, t_0)\,\boldsymbol{v}(t), \tag{6.35}$$

where we are seeking the vector $\boldsymbol{v}$. Substituting (6.35) into (6.33), we have

$$\begin{aligned}
\boldsymbol{x}_{pi} - \mathbf{A}\boldsymbol{x}_{pi} &= \frac{d}{dt}\left(\boldsymbol{\Phi}(t, t_0)\,\boldsymbol{v}\right) - \mathbf{A}\boldsymbol{\Phi}(t, t_0)\,\boldsymbol{v}, \\
&= \frac{d\boldsymbol{\Phi}(t, t_0)}{dt}\boldsymbol{v} + \boldsymbol{\Phi}(t, t_0)\frac{d\boldsymbol{v}}{dt} - \mathbf{A}\boldsymbol{\Phi}(t, t_0)\,\boldsymbol{v}, \\
&= \boldsymbol{\Phi}(t, t_0)\frac{d\boldsymbol{v}}{dt}, \qquad (6.36) \\
\Rightarrow \frac{d\boldsymbol{v}}{dt} &= \boldsymbol{\Phi}(t, t_0)^{-1}\boldsymbol{u}, \qquad (6.37)
\end{aligned}$$

where we know that the inverse of the state transition matrix always exists. Therefore, integrating (6.37) from t_0 to t yields

$$\boldsymbol{v}(t) - \boldsymbol{v}(t_0) = \int_{t_0}^{t} \boldsymbol{\Phi}^{-1}(s, t_0)\,\mathbf{B}\boldsymbol{u}(s)\,ds. \tag{6.38}$$

Given the initial conditions requirement for the particular integral, we can infer that if $\boldsymbol{x}_{pi}(t_0) = 0$, then $\boldsymbol{v}(t_0) = 0$. Therefore we have that the particular integral is

$$\begin{aligned}\boldsymbol{x}_{pi}(t) &= \boldsymbol{\Phi}(t, t_0)\,\boldsymbol{v}(t),\\ &= \boldsymbol{\Phi}(t, t_0)\int_{t_0}^{t} \boldsymbol{\Phi}^{-1}(s, t_0)\,\mathbf{B}\boldsymbol{u}(s)\,ds,\\ &= \int_{t_0}^{t} \boldsymbol{\Phi}(t, s)\,\mathbf{B}\boldsymbol{u}(s)\,ds. \end{aligned} \tag{6.39}$$

Thus, the complete general solution to the controlled problem is given by

$$\begin{aligned}\boldsymbol{x}(t) &= \boldsymbol{x}_u(t) + \boldsymbol{x}_{pi}(t),\\ &= \boldsymbol{\Phi}(t, t_0)\,\boldsymbol{x}(t_0) + \int_{t_0}^{t} \boldsymbol{\Phi}(t, s)\,\mathbf{B}\boldsymbol{u}(s)\,ds. \end{aligned} \tag{6.40}$$

To illustrate how to solve a controlled problem we return to Example 2, where the state transition matrix for this problem has been shown through multiple approaches to be

$$\boldsymbol{\Phi}(t, t_0) = \begin{pmatrix} 1 & t - t_0 \\ 0 & 1 \end{pmatrix}. \tag{6.41}$$

We know that the control for this example only affects x_2, therefore $\mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. We are going to assume that $t_0 = 0$, that $\boldsymbol{x}_0 = \mathbf{0}$, and finally that the control is $u(t) = t$. This results in

$$\begin{aligned}\boldsymbol{x}(t) &= \int_0^t \begin{pmatrix} 1 & t - s \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} s\,ds,\\ &= \int_0^t \begin{pmatrix} t - s \\ 1 \end{pmatrix} s\,ds,\\ &= \int_0^t \begin{pmatrix} st - s^2 \\ s \end{pmatrix} ds,\\ &= \left[\begin{pmatrix} \frac{s^2}{2}t - \frac{s^3}{3} \\ \frac{s^2}{2} \end{pmatrix}\right]_0^t,\\ &= \begin{pmatrix} \frac{t^3}{6} \\ \frac{t^2}{2} \end{pmatrix}. \end{aligned} \tag{6.42}$$

6.3.1 Controllability

Over the two previous sections and subsections, we have been able to derive techniques to the uncontrolled and controlled problem. The solutions that we have found are valid for many different types of $\mathbf{A}$ and for all $\mathbf{B}$ matrices, and are how we can see the response of the system to the control $\boldsymbol{u}$. We now introduce the concept of **controllability**, with the following definition for completely controllable.

Definition 6.8. The system of differential equations $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, is said to be ***completely controllable*** if for any t_0, any $\boldsymbol{x}_0$ and $\boldsymbol{x}_f$, there exists a final time $t_f > t_0$ and control $\boldsymbol{u}(t)$ with $t \in [t_0, t]$ such that $\boldsymbol{x}(t_f) = \boldsymbol{x}_f$.

The *completely* component of this newly defined property implies that the definition must hold for all possible conditions and for all possible final time states. It is also assumed that the control that is used in this definition is piecewise continuous.

Completely controllability can be guaranteed in advance by applying certain checks to the **A** and **B** matrices. There are two possible cases that stand out:

1. **A** and **B** are constant.
2. **A** and **B** vary with time.

Considering the constant term first, there is a theorem that enables us to know in advance if the system is completely controllable.

Theorem 6.9. *If the **A** and **B** matrices are constant matrices, then the system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$ is completely controllable if and only if the **controllability matrix***

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{B} & \boldsymbol{AB} & \boldsymbol{A}^2\boldsymbol{B} & \cdots & \boldsymbol{A}^{n-1}\boldsymbol{B} \end{bmatrix} \tag{6.43}$$

has rank n.

An important feature to note here is that the controllability matrix is a $n \times nm$ matrix.

We shall present two short examples now to illustrate how we use the controllability matrix to inform us if the system is controllable.

Example 6.10. *Given* $\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$ *and* $\boldsymbol{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, *is the resulting control system completely controllable?*

As we have just stated above in Theorem 6.9, if we can show that the controllability matrix has full rank then thus the system is completely controllable. To form the controllability matrix, we require **AB**, which is

$$\mathbf{AB} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Therefore, the controllability matrix for this example is $\mathbf{U} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$; we can easily see that this matrix has $rank(\mathbf{U}) = 2$, which implies that the matrix has full rank and the system is completely controllable.

If we look at the equivalent linear equations that matrices **A** and **B** represent, we see that while x_1 is a function of x_2, we can control x_2 and therefore by association we can control x_1 as well.

Example 6.11. *Given the system of linear ordinary differential equations given by*

$$\begin{aligned} \dot{x}_1 &= x_1 + 3x_2 + u, \\ \dot{x}_2 &= x_2, \end{aligned} \tag{6.44}$$

determine if the control system is completely controllable.

We need to form the controllability matrix, which means that we need to determine **A** and **B** and hence **U**:

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \mathbf{AB} = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$
$$\Rightarrow \mathbf{U} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

We can clearly see that the controllability matrix above is $rank\,(\mathbf{U}) = 1$ and is therefore rank deficient, which then implies that this system is not completely controllable. If we look at the system defined in (6.44), we can see that only x_1 is affected by the control u, yet x_2 is not a function of x_1 and is therefore not affected by changes in the control and its effect on x_1.

Exercise 6.12. *For what values of a, b, c, and d is the control system*

$$\dot{x}_1 = ax_1 + bx_2 + u,$$
$$\dot{x}_2 = cx_1 + dx_2$$

completely controllable?

Exercise 6.13. *For what values of a and b is the control system*

$$\dot{x}_1 = x_2 + au,$$
$$\dot{x}_2 = x_1 + bu$$

completely controllable?

We now return to Example 3, the oscillating spring, to determine if this system is completely controllable.

Example 6.14. *Given the second-order ordinary differential equation*

$$\ddot{z} + \omega^2 z = u,$$

determine if the equivalent system of linear ordinary differential equations are completely controllable.

Solution. Introduce the change of variable x_1 and x_2 such that

$$\left.\begin{aligned} x_1 &= z \\ x_2 &= \dot{z} \end{aligned}\right\} \Rightarrow \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\omega^2 x_1 + u, \end{aligned}$$

which gives us

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -\omega^2 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \mathbf{AB} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Therefore, the controllability matrix is

$$\mathbf{U} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

which can clearly be seen to have full rank and therefore this control system is completely controllable.

We now move on to consider the time-varying case where $\mathbf{A} = \mathbf{A}(t)$ and $\mathbf{B} = \mathbf{B}(t)$ and are such that each component of these matrices is a continuous function of t for $t > t_0$. Given that $\mathbf{A}$ and $\mathbf{B}$ are time varying, we require an equivalent measure to determine whether or not these types of systems are completely controllable. The equivalent measure is defined in the following theorem:

Theorem 6.15. *Consider the system* $\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(t)\boldsymbol{x}(t) + \boldsymbol{Bu}(t)$ *for* $t > t_0$, *if the matrix*

$$\boldsymbol{M} = \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0, s)\boldsymbol{B}(s)\boldsymbol{B}^T(s)\boldsymbol{\Phi}^T(t_0, s)\,ds, \tag{6.45}$$

is non-singular for some $t_f > t_0$ *then the system is completely controllable and the control is given by*

$$\boldsymbol{u}(t) = -\boldsymbol{B}^T(t)\boldsymbol{\Phi}(t_0, t)^T\boldsymbol{M}^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{\Phi}\left(t_0, t_f\right)\boldsymbol{x}_f\right), \tag{6.46}$$

which moves the state from $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ *to* $\boldsymbol{x}\left(t_f\right) = \boldsymbol{x}_f$ *for any* $\boldsymbol{x}_f$.

Conversely, if the system is completely controllable so that for any initial states $\boldsymbol{x}_0$ *and any final state* $\boldsymbol{x}_f$, *there exists a* t_f *such that the system can be controlled from* $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ *to* $\boldsymbol{x}\left(t_f\right) = \boldsymbol{x}_f$, *then* $\boldsymbol{M}$ *defined in* (6.45) *must be non-singular.*

Proof. Suppose that $\mathbf{M}$ is non-singular, then the control defined in (6.46) exists and the response of the system is

$$\begin{aligned}\boldsymbol{x}(t) &= \boldsymbol{\Phi}(t, t_0)\left\{\boldsymbol{x}_0 + \int_{t_0}^{t} \boldsymbol{\Phi}(t_0, s)\mathbf{B}(s)\boldsymbol{u}(s)\,ds\right\},\\ &= \boldsymbol{\Phi}(t, t_0)\left\{\boldsymbol{x}_0 + \int_{t_0}^{t} \boldsymbol{\Phi}(t_0, s)\mathbf{B}(s)\left(-\mathbf{B}^T(s)\boldsymbol{\Phi}(t_0, s)^T\mathbf{M}^{-1}\boldsymbol{c}\right)ds\right\},\end{aligned}$$

where $\boldsymbol{c} \equiv \boldsymbol{x}_0 - \boldsymbol{\Phi}\left(t_0, t_f\right)\boldsymbol{x}_f$ is a constant vector. Now, evaluating the expression above at $t = t_f$, we have

$$\begin{aligned}\boldsymbol{x}\left(t_f\right) &= \boldsymbol{\Phi}\left(t_f, t_0\right)\left\{\boldsymbol{x}_0 - \mathbf{M}\mathbf{M}^{-1}\boldsymbol{c}\right\},\\ &= \boldsymbol{\Phi}\left(t_f, t_0\right)\boldsymbol{\Phi}\left(t_0, t_f\right)\boldsymbol{x}_f = \boldsymbol{x}_f.\end{aligned}$$

There have been no assumptions made on $\boldsymbol{x}_0$ or $\boldsymbol{x}_f$ and therefore this derivation is valid for all $\boldsymbol{x}_0$, $\boldsymbol{x}_f$, t_0, and t_f and therefore the system is completely controllable.

To prove the converse statement in Theorem 6.15, we assume that the system is completely controllable, thus the system may be controlled to any state by time $t = t_f$. Therefore we shall fix t_f and assume that $\mathbf{M}$ is singular. By this assumption there must exist a vector $\boldsymbol{w} \neq \mathbf{0}$ such that $\mathbf{M}\boldsymbol{w} = \mathbf{0}$ and therefore $\boldsymbol{w}^T\mathbf{M}\boldsymbol{w} = 0$. Constructing $\boldsymbol{w}^T\mathbf{M}\boldsymbol{w}$, we have

$$\begin{aligned}\boldsymbol{w}^T\mathbf{M}\boldsymbol{w} &= \int_{t_0}^{t_f} \boldsymbol{w}^T\boldsymbol{\Phi}(t_0, s)\mathbf{B}(s)\mathbf{B}^T(s)\boldsymbol{\Phi}^T(t_0, s)\boldsymbol{w}\,ds,\\ &\equiv \int_{t_0}^{t_f} \boldsymbol{r}(s)^T\boldsymbol{r}(s)\,ds,\end{aligned}$$

where $\boldsymbol{r}(s) = \mathbf{B}^T(s)\boldsymbol{\Phi}^T(t_0, s)\boldsymbol{w}$ is a continuous real function of s. For $\boldsymbol{w}^T\mathbf{M}\boldsymbol{w} = 0$ we must have

$$\boldsymbol{r}(s) = \mathbf{B}^T(s)\boldsymbol{\Phi}^T(t_0, s)\boldsymbol{w} = \mathbf{0}, \quad \forall s \in \left[t_0, t_f\right].$$

Now it has been assumed that the system is completely controllable so therefore there exists a control $\boldsymbol{u} = \boldsymbol{u}(t)$ that takes the system from the initial time and state $\boldsymbol{x}(t_0) = \mathbf{0}$ to $\boldsymbol{x}(t_f) = \boldsymbol{\Phi}(t_f, t_0)\boldsymbol{w}$. Therefore constructing the final state gives

$$\begin{aligned}\boldsymbol{x}(t_f) &= \boldsymbol{\Phi}(t_f, t_0)\boldsymbol{w} = \boldsymbol{\Phi}(t_f, t_0)\int_{t_0}^{t_f},\\ &\Rightarrow \boldsymbol{w}^T\boldsymbol{w} = 0,\end{aligned}$$

which is a contradiction. Thus by proof by contradiction $\mathbf{M}$ must be non-singular.

Example 6.16. *For the control system*

$$\begin{aligned}\dot{x}_1 &= x_2,\\ \dot{x}_2 &= -x_1 + u,\end{aligned} \tag{6.47}$$

construct the controllability matrix $\mathbf{M}$ *on the interval* $[0, \pi]$. *Hence find a control* u *that moves the system from* $\boldsymbol{x}_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ *at* $t = 0$ *to* $\boldsymbol{x}(\pi) = \begin{pmatrix} 0 \\ -\frac{\pi}{2} \end{pmatrix}$ *at* $t = \pi$.

The first step is to find the state transition matrix $\boldsymbol{\Phi}$. We consider the differential equation $\ddot{x}_1 = \dot{x}_2 = -x_1$ which has the general solution of the form $x_1(t) = A\cos t + B\sin t$. Differentiating this general solution for x_1 results in $x_2(t) = -A\sin t + B\cos t$. Using the unit vector initial conditions for the fundamental solutions results in

$$\begin{aligned}\boldsymbol{\phi}_1(0) &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \Rightarrow A = 1,\ B = 0,\\ \boldsymbol{\phi}_2(0) &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} \Rightarrow A = 0,\ B = 1.\end{aligned}$$

Therefore the state transition matrix for this example is

$$\begin{aligned}\boldsymbol{\Phi}(t, 0) &= \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix},\\ \boldsymbol{\Phi}(t, t_0) &= \begin{pmatrix} \cos(t - t_0) & \sin(t - t_0) \\ -\sin(t - t_0) & \cos(t - t_0) \end{pmatrix}.\end{aligned}$$

The next step is to construct the controllability matrix, $\mathbf{M}$,

$$\begin{aligned}\mathbf{M} &= \int_0^{\pi} \boldsymbol{\Phi}(0, s)\begin{pmatrix} 0 \\ 1 \end{pmatrix}\begin{pmatrix} 0 & 1 \end{pmatrix}\boldsymbol{\Phi}^T(0, s)\,ds,\\ &= \int_0^{\pi} \begin{pmatrix} \cos s & \sin s \\ -\sin s & \cos s \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix}\begin{pmatrix} 0 & 1 \end{pmatrix}\begin{pmatrix} \cos s & -\sin s \\ \sin s & \cos s \end{pmatrix} ds,\\ &= \int_0^{\pi} \begin{pmatrix} -\sin s \\ \cos s \end{pmatrix}\begin{pmatrix} -\sin s & \cos s \end{pmatrix} ds,\\ &= \int_0^{\pi} \begin{pmatrix} \sin^2 s & -\sin s\cos s \\ -\sin s\cos s & \cos^2 s \end{pmatrix} ds,\\ &= \begin{pmatrix} \frac{\pi}{2} & 0 \\ 0 & \frac{\pi}{2} \end{pmatrix} \Rightarrow \mathbf{M}^{-1} = \begin{pmatrix} \frac{2}{\pi} & 0 \\ 0 & \frac{2}{\pi} \end{pmatrix}.\end{aligned}$$

The final step is to construct the control u:

$$\begin{aligned} u(t) &= -\mathbf{B}^T \boldsymbol{\Phi}(0,t)\,\mathbf{M}^{-1}\left\{\begin{pmatrix} 0 \\ 0 \end{pmatrix} - \boldsymbol{\Phi}(0,\pi)\begin{pmatrix} 0 \\ -\frac{\pi}{2} \end{pmatrix}\right\}, \\ &= \begin{pmatrix} 0 & 1 \end{pmatrix}\begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix}\mathbf{M}^{-1}\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} 0 \\ -\frac{\pi}{2} \end{pmatrix}, \\ &= -\frac{2}{\pi}\begin{pmatrix} -\sin t & \cos t \end{pmatrix}\begin{pmatrix} 0 \\ -\frac{\pi}{2} \end{pmatrix}, \\ u(t) &= \cos t. \end{aligned}$$

Exercise 6.17. *Determine if the following control system are completely controllable:*

$$\dot{\boldsymbol{x}} = \begin{pmatrix} -2 & 2 \\ 1 & -1 \end{pmatrix}\boldsymbol{x} + \begin{pmatrix} 1 \\ 0 \end{pmatrix}u,$$

$$\dot{\boldsymbol{x}} = \begin{pmatrix} 1 & 2 & -1 \\ 0 & 2 & 0 \\ 1 & -4 & 3 \end{pmatrix}\boldsymbol{x} + \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}u.$$

We now consider an important feature of control theory, which is known as **equivalence**.

6.3.2 Equivalence

We start by considering two systems, S_1 and S_2, defined by

$$\begin{aligned} S_1: &\quad \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \quad \boldsymbol{y} = \mathbf{C}\boldsymbol{x}, \\ S_2: &\quad \dot{\hat{\boldsymbol{x}}} = \hat{\mathbf{A}}\hat{\boldsymbol{x}} + \hat{\mathbf{B}}\boldsymbol{u}, \quad \boldsymbol{y} = \hat{\mathbf{C}}\hat{\boldsymbol{x}}, \end{aligned} \tag{6.48}$$

where both systems have the same control, $\boldsymbol{u}$, and the same output, $\boldsymbol{y}$.

The systems in (6.48) are said to be equivalent if and only if there exists an invertible $n \times n$ matrix $\mathbf{P} = \mathbf{P}(t)$ such that $\hat{\boldsymbol{x}} = \mathbf{P}\boldsymbol{x}$. Therefore, given this new matrix, we consider finding expressions for $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, and $\hat{\mathbf{C}}$:

$$\begin{aligned} \frac{d}{dt}\hat{\boldsymbol{x}} &= \frac{d}{dt}(\mathbf{P}\boldsymbol{x}) = \dot{\mathbf{P}}\boldsymbol{x} + \mathbf{P}\dot{\boldsymbol{x}}, \\ &= \dot{\mathbf{P}}\boldsymbol{x} + \mathbf{P}(\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}), \\ &= (\dot{\mathbf{P}} + \mathbf{P}\mathbf{A})\boldsymbol{x} + \mathbf{P}\mathbf{B}\boldsymbol{u}, \\ &= (\dot{\mathbf{P}} + \mathbf{P}\mathbf{A})\mathbf{P}^{-1}\hat{\boldsymbol{x}} + \mathbf{P}\mathbf{B}\boldsymbol{u}, \qquad (6.49) \\ \boldsymbol{y} &= \mathbf{C}\boldsymbol{x} = \mathbf{C}\mathbf{P}^{-1}\boldsymbol{x}. \qquad (6.50) \end{aligned}$$

Given the expressions in (6.49) and (6.50), we can say that control systems S_1 and S_2 are equivalent if

$$\hat{\mathbf{A}} \equiv (\dot{\mathbf{P}} + \mathbf{P}\mathbf{A})\mathbf{P}^{-1}, \tag{6.51}$$

$$\hat{\mathbf{B}} \equiv \mathbf{P}\mathbf{B}, \tag{6.52}$$

$$\hat{\mathbf{C}} \equiv \mathbf{C}\mathbf{P}^{-1}. \tag{6.53}$$

With the expressions established in (6.51) that link the matrices in control system S_1 to control system S_2, it is possible to state the following theorem and corollary that links the state transition matrices between the two control systems and hence complete controllability.

Theorem 6.18. *If $\mathbf{\Phi}(t, t_0)$ is the state transition matrix for control system S_1 then the state transition matrix for the equivalent control system S_2 is*

$$\hat{\mathbf{\Phi}}(t, t_0) \equiv \boldsymbol{P}(t_0)\,\mathbf{\Phi}(t, t_0)\,\boldsymbol{P}(t_0)^{-1}. \tag{6.54}$$

Proof. To prove Theorem 6.18, we need to show that $\hat{\mathbf{\Phi}}(t, t_0)$ satisfies

$$\frac{d}{dt}\hat{\mathbf{\Phi}}(t, t_0) = \hat{\mathbf{A}}\hat{\mathbf{\Phi}}(t, t_0),$$
$$\hat{\mathbf{\Phi}}(t_0, t_0) = \mathbf{I}.$$

Taking the second property first, we have that

$$\hat{\mathbf{\Phi}}(t_0, t_0) = \mathbf{P}(t_0)\,\mathbf{\Phi}(t_0, t_0)\,\mathbf{P}(t_0)^{-1} = \mathbf{I}.$$

The first property is proven as follows:

$$\begin{aligned}
\frac{d}{dt}\left(\mathbf{\Phi}(t, t_0)\right) &= \frac{d}{dt}\left(\mathbf{P}(t)\,\mathbf{\Phi}(t, t_0)\,\mathbf{P}(t_0)^{-1}\right), \\
&= \dot{\mathbf{P}}(t)\,\mathbf{\Phi}(t, t_0)\,\mathbf{P}(t_0)^{-1} + \mathbf{P}(t)\frac{d}{dt}\mathbf{\Phi}(t, t_0)\,\mathbf{P}(t_0)^{-1}, \\
&= \left(\dot{\mathbf{P}}(t) + \mathbf{P}(t)\,\mathbf{A}\right)\mathbf{\Phi}(t, t_0)\,\mathbf{P}(t_0)^{-1}, \\
&= \left(\dot{\mathbf{P}}(t) + \mathbf{P}(t)\,\mathbf{A}\right)\mathbf{P}(t)^{-1}\,\mathbf{P}(t)\,\mathbf{\Phi}(t, t_0)\,\mathbf{P}(t_0)^{-1}, \\
&= \hat{\mathbf{A}}\hat{\mathbf{\Phi}}(t, t_0),
\end{aligned}$$

which proves the first property of Theorem 6.18.

As an extension to Theorem 6.18, we have the following corollary.

Corollary 6.19. *If control systems S_1 and S_2 are equivalent control systems, then S_1 is completely controllable if and only if S_2 is completely controllable.*

Proof. Given that the control system S_2 is completely controllable, then it must have a controllability matrix $\hat{\mathbf{M}}$ which is defined by

$$\begin{aligned}
\hat{\mathbf{M}} &= \int_{t_0}^{t_f} \hat{\mathbf{\Phi}}(t_0, s)\,\hat{\mathbf{B}}\hat{\mathbf{B}}\hat{\mathbf{\Phi}}^T(t_0, s)\,ds, \\
&= \int_{t_0}^{t_f} \left(\mathbf{P}(t_0)\,\mathbf{\Phi}(t_0, s)\,\mathbf{P}(s)^{-1}\right)\left(\mathbf{P}(s)\,\mathbf{B}\right)\left(\mathbf{B}^T\mathbf{P}(s)^T\right)\left(\mathbf{P}(s)^{-T}\,\mathbf{\Phi}^T(t_0, s)\,\mathbf{P}(t_0)^T\right)ds, \\
&= \int_{t_0}^{t_f} \mathbf{P}(t_0)\,\mathbf{\Phi}(t_0, s)\,\mathbf{B}\mathbf{B}^T\mathbf{\Phi}^T(t_0, s)\,\mathbf{P}(t_0)^T\,ds, \\
&= \mathbf{P}(t_0)\,\mathbf{M}\mathbf{P}(t_0),
\end{aligned}$$

where $\mathbf{M}$ is the controllability matrix for control system S_1. Therefore $\mathbf{M}$ is non-singular if and only if $\hat{\mathbf{M}}$ is non-singular, which proves the corollary.

The importance of equivalence will become apparent when we move on to duality, but first we consider another property of control systems which will lay the basis for linking certain data assimilation methods back to control theory: observability.

6.4 Observability

Consider the control system

$$S: \begin{cases} \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \\ \boldsymbol{y} = \mathbf{C}\boldsymbol{x}, \end{cases} \tag{6.55}$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ could all be functions of t. The question becomes: given output measurements $\boldsymbol{y}$, is it possible to reconstruct the state of the system, $\boldsymbol{x}$?

To address this question we introduce the definition of completely observable:

Definition 6.20. A control system is said to be completely observable if for any initial time t_0 and any initial conditions at that time, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, there exists a finite time t_f such that given the control $\boldsymbol{u}(t)$ and output measurements $\boldsymbol{y}(t)$ for $t \in [t_0, t_f]$ it is possible to reconstruct the state $\boldsymbol{x}_0$.

As with controllability, there is a matrix (measure) that enables the user to know if the system is completely observable or not. The matrix is referred to as the **observability matrix** and has the following theorem defining its property:

Theorem 6.21. *Given the control system S defined above with $\boldsymbol{A} = \boldsymbol{A}(t)$ and $\boldsymbol{C} = \boldsymbol{C}(t)$ then the control system, S, is completely observable if and only if*

$$\boldsymbol{N} = \int_{t_0}^{t_f} \boldsymbol{\Phi}^T(s, t_0)\boldsymbol{C}^T\boldsymbol{C}\boldsymbol{\Phi}(s, t_0)\, ds, \tag{6.56}$$

is non-singular.

Proof. The starting point for the first part of the proof is to assume that $\mathbf{N}$ is invertible and then show that it is possible to construct $\boldsymbol{x}_0$. To simplify the proof we only consider the case where $\boldsymbol{u}(t) = 0$, which does not affect the proof except to remove an additional term that is easy to deal with.

The response to the control system S is $\boldsymbol{x}(t) = \boldsymbol{\Phi}(t, t_0)\boldsymbol{x}_0$ and therefore the output measurement can be expressed as $\boldsymbol{y}(t) \equiv \mathbf{C}(t)\,\boldsymbol{\Phi}(t, t_0)\boldsymbol{x}_0$.

Consider the following integral:

$$\begin{aligned} \int_{t_0}^{t_f} \boldsymbol{\Phi}^T(s, t_0)\mathbf{C}^T\boldsymbol{y}(t)\, ds &= \int_{t_0}^{t_f} \boldsymbol{\Phi}^T(s, t_0)\mathbf{C}^T\mathbf{C}\boldsymbol{\Phi}(s, t_0)\boldsymbol{x}_0 ds, \\ &= \mathbf{N}\boldsymbol{x}_0. \end{aligned}$$

Therefore, inverting $\mathbf{N}$ results in the expression for the initial conditions as

$$\boldsymbol{x}_0 = \mathbf{N}^{-1} \int_{t_0}^{t_f} \boldsymbol{\Phi}^T(s, t_0)\mathbf{C}^T\boldsymbol{y}(t)\, ds, \tag{6.57}$$

which is possible to evaluate and hence this proves that if the observability matrix $\mathbf{N}$ is invertible, then the control system is completely observable.

The next step is to prove that if the control system is completely observable, then $\mathbf{N}$ must not be singular. Therefore, if $\mathbf{N}$ is singular, then there exists a non-zero vector $\boldsymbol{w}$ such that $\mathbf{N}\boldsymbol{w} = \mathbf{0}$. As with the proof for the invertibility of the controllability matrix, we consider

$$\begin{aligned}\boldsymbol{w}^T \mathbf{N} \boldsymbol{w} &= \int_{t_0}^{t_f} \boldsymbol{w}^T \boldsymbol{\Phi}^T (s, t_0)\, \mathbf{C}^T \mathbf{C} \boldsymbol{\Phi} (s, t_0)\, \boldsymbol{w} ds, \\ &\equiv \int_{t_0}^{t_f} \boldsymbol{z}^T (s)\, \boldsymbol{z} (s)\, ds = 0.\end{aligned}$$

Now the vector $\boldsymbol{z}(s)$ is a continuous function of s and therefore $\boldsymbol{z}(s) = \mathbf{0}$ for all $s \in (t_0, t_f)$, thus

$$\mathbf{C}\boldsymbol{\Phi}(s, t_0)\, \boldsymbol{w} = \mathbf{0}, \quad \forall s \in (t_0, t_f).$$

It has been assumed that the control system is completely observable, therefore given that the output measurement is defined as $\boldsymbol{y} = \mathbf{C}\boldsymbol{x}$ and the output from $\boldsymbol{x}_0 = \boldsymbol{w}$, then $\boldsymbol{y}(t) = \mathbf{0}$. It is known that $\mathbf{C}$ is non-zero and that the state transition matrix is not equal to zero $\forall s \in (t_0, t_f)$; therefore $\boldsymbol{w} = \mathbf{0}$. However, $\boldsymbol{w} \neq \mathbf{0}$ and it follows that our output is $\boldsymbol{y}(t) = \mathbf{0}$, which does not give sufficient information to reconstruct $\boldsymbol{x}_0$. This is a contradiction and therefore $\mathbf{N}$ is non-singular.

During both parts of the proof there have been no assumptions made about t_0 and $\boldsymbol{x}_0$, and therefore the proof holds for any values.

An important property that arose during the first part of the proof of Theorem 6.21 is an expression that enables the reconstruction of the initial conditions for the control system if the system is completely observable; see (6.57).

Example 6.22. *Consider the control system*

$$\begin{aligned}\dot{x}_1 &= 2x_2, \\ \dot{x}_2 &= u,\end{aligned}$$

where u is the control, with output $y = x_1$, on the interval $[0, 1]$. Given that the measured output response to the control $u \equiv 0$ is $y(t) = 4 - 4t$ for $t \in [0, 1]$, find $x_1(0)$ and $x_2(0)$.

First identifying $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, we have

$$\mathbf{A} = \begin{pmatrix} 0 & 2 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 0 & 1 \end{pmatrix}.$$

Given that $\mathbf{A}$ is a constant matrix we can use the power series approximation to the state transition matrix $e^{t\mathbf{A}}$, but we are also fortunate that $\mathbf{A}^2 = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, which then makes

$$e^{t\mathbf{A}} = \mathbf{I} + t\mathbf{A} = \begin{pmatrix} 1 & 2t \\ 0 & 1 \end{pmatrix}.$$

The next step is to construct the observability matrix

$$\mathbf{N} = \int_0^1 \boldsymbol{\Phi}^T (s, 0)\, \mathbf{C}^T \mathbf{C} \boldsymbol{\Phi} (s, 0)\, ds,$$

$$
\begin{aligned}
&= \int_0^1 \begin{pmatrix} 1 & 0 \\ 2s & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 2s \\ 0 & 1 \end{pmatrix} ds, \\
&= \int_0^1 \begin{pmatrix} 1 \\ 2s \end{pmatrix} \begin{pmatrix} 1 & 2s \end{pmatrix} ds = \int_0^1 \begin{pmatrix} 1 & 2s \\ 2s & 4s^2 \end{pmatrix} ds, \\
&= \begin{pmatrix} 1 & 1 \\ 1 & \frac{4}{3} \end{pmatrix}.
\end{aligned}
$$

It is the inverse of the observability matrix that is required, which can easily be verified to be

$$
\mathbf{N}^{-1} = \begin{pmatrix} 4 & -3 \\ -3 & 3 \end{pmatrix}.
$$

Evaluating (6.57) for this example gives

$$
\begin{aligned}
\boldsymbol{x}(0) &= \mathbf{N}^{-1} \int_0^1 \begin{pmatrix} 1 \\ 2s \end{pmatrix} \begin{pmatrix} 4 - 4s \end{pmatrix} ds = \mathbf{N}^{-1} \int_0^1 \begin{pmatrix} 4 - 4s \\ 8s - 8s^2 \end{pmatrix} ds, \\
&= \mathbf{N}^{-1} \left[\begin{pmatrix} 4 - 2s^2 \\ 4s^2 - \frac{8}{3}s^3 \end{pmatrix} \right]_0^1 = \mathbf{N}^{-1} \begin{pmatrix} 2 \\ \frac{4}{3} \end{pmatrix}, \\
&= \begin{pmatrix} 4 & -3 \\ -3 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ \frac{4}{3} \end{pmatrix} = \begin{pmatrix} 4 \\ -2 \end{pmatrix}.
\end{aligned}
$$

Exercise 6.23. *For the following control system,*

$$
\dot{\boldsymbol{x}} = \begin{pmatrix} -2 & 2 \\ 1 & -1 \end{pmatrix} \boldsymbol{x} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u,
$$

determine if any of the following measurements make the control system completely observable:

$$
y_1 = \begin{pmatrix} 1 & 1 \end{pmatrix} \boldsymbol{x}, \quad y_2 = \begin{pmatrix} 0 & 1 \end{pmatrix} \boldsymbol{x}, \quad y_3 = \begin{pmatrix} 1 & 0 \end{pmatrix} \boldsymbol{x}, \quad y_4 = \begin{pmatrix} 1 & -1 \end{pmatrix} \boldsymbol{x}.
$$

Now that we have the definition for completely controllable, completely observable, and equivalent systems, we consider how to combine all of these properties together; this comes through **duality**.

6.5 Duality

While it may not be obvious how completely controllability and complete observability are linked, it is possible, as we shall see how with the proof of the following theorem.

Theorem 6.24. *The control system*

$$
S_1 = \begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} \end{cases} \qquad (t > t_0) \tag{6.58}
$$

is completely controllable (completely observable) if and only if the control system

$$S_2 = \begin{cases} \dot{\boldsymbol{x}} = -\boldsymbol{A}^T\boldsymbol{x} + \boldsymbol{C}^T\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{B}^T\boldsymbol{x} \end{cases} \qquad (t > t_0) \tag{6.59}$$

is completely observable (completely controllable).

Proof. The foundation of the proof of Theorem 6.24 is to prove that the controllability matrix $\mathbf{M}$ (observability matrix $\mathbf{N}$) of control system S_1 is equivalent to the observability matrix $\mathbf{N}$ (controllability matrix $\mathbf{M}$) of control system S_2.

Now consider the following two problems for finding state transition matrices:

$$\begin{array}{ll} \frac{d\boldsymbol{\Phi}}{dt} = \mathbf{A}\boldsymbol{\Phi} \qquad (t > t_0) & \frac{d\hat{\boldsymbol{\Phi}}}{dt} = -\mathbf{A}^T\hat{\boldsymbol{\Phi}} \quad (t > t_0) \\ \boldsymbol{\Phi}(t_0, t_0) = \mathbf{I} & \hat{\boldsymbol{\Phi}}(t_0, t_0) = \mathbf{I} \end{array}$$

Given these two systems, we need to show that $\hat{\boldsymbol{\Phi}}(t, t_0) \equiv -\boldsymbol{\Phi}^T(t, t_0)$. This is achieved through using the trick that $\mathbf{X}\mathbf{X}^{-1} = \mathbf{I}$ and that it follows that for invertible matrices,

$$\frac{d}{dt}\left(\mathbf{X}\mathbf{X}^{-1}\right) = \mathbf{0}.$$

Applying the product rule for differentiation yields

$$\begin{aligned} \mathbf{X}\frac{d}{dt}\left(\mathbf{X}^{-1}\right) &= -\frac{d\mathbf{X}}{dt}\mathbf{X}^{-1}, \\ \frac{d}{dt}\left(\mathbf{X}^{-1}\right) &= -\mathbf{X}^{-1}\frac{d\mathbf{X}}{dt}\mathbf{X}^{-1}. \end{aligned}$$

Applying this expression to the state transition matrix, $\boldsymbol{\Phi}^T(t, t_0)$, results in

$$\begin{aligned} \frac{d}{dt}\left(\boldsymbol{\Phi}^T\right) &= -\left(\boldsymbol{\Phi}^{-1}\frac{\mathrm{d}\boldsymbol{\Phi}}{dt}\boldsymbol{\Phi}^{-1}\right)^T, \\ &= -\left(\boldsymbol{\Phi}^{-1}\mathbf{A}\boldsymbol{\Phi}\boldsymbol{\Phi}^{-1}\right)^T, \\ &= -\mathbf{A}\boldsymbol{\Phi}^{-T}, \end{aligned}$$

and clearly the condition for the state transition matrix at the initial time $\hat{\boldsymbol{\Phi}}(t_0, t_0) = \boldsymbol{\Phi}^{-T}(t_0, t_0) = \mathbf{I}$ is satisfied and so we have shown that $\hat{\boldsymbol{\Phi}} \equiv \boldsymbol{\Phi}^{-T}$.

The final step of the proof is to consider the controllability matrix $\mathbf{M}$ for control system S_1,

$$\mathbf{M} = \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0, s)\,\mathbf{B}\mathbf{B}^T\boldsymbol{\Phi}^T(t_0, s)\,ds,$$

and the observability matrix $\hat{\mathbf{N}}$ for control system S_2:

$$\begin{aligned} \hat{\mathbf{N}} &= \int_{t_0}^{t_f} \hat{\boldsymbol{\Phi}}^T(s, t_0)\,\mathbf{B}\mathbf{B}^T\hat{\boldsymbol{\Phi}}^T(s, t_0)\,ds, \\ &= \int_{t_0}^{t_f} \boldsymbol{\Phi}^{-1}(s, t_0)\,\mathbf{B}\mathbf{B}^T\boldsymbol{\Phi}^{-T}(s, t_0)\,ds, \end{aligned}$$

$$= \int_{t_0}^{t_f} \mathbf{\Phi}(t_0, s)\,\mathbf{B}\mathbf{B}^T\,\mathbf{\Phi}^T(t_0, s)\,ds,$$
$$= \mathbf{M}.$$

Since the result is symmetric in control systems S_1 and S_2, this completes the proof of Theorem 6.24.

An important by-product of this duality results is an easily provable observability condition for the time-invariant case. It has already been shown that if the matrix $\mathbf{V} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix}$ has full rank, then the associated control system is completely controllable. We now extend this for a condition for completely observable.

Consider the control system $\begin{matrix} \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u} \\ \boldsymbol{y} = \mathbf{C}\boldsymbol{x} \end{matrix}$ which is known to be completely observable if and only if the dual system $\begin{matrix} \dot{\boldsymbol{x}} = -\mathbf{A}\boldsymbol{x} + \mathbf{C}^T\boldsymbol{u} \\ \boldsymbol{y} = \mathbf{B}^T\boldsymbol{x} \end{matrix}$ is completely controllable. Therefore, the dual system is completely controllable if and only if the matrix $\mathbf{U} = \begin{bmatrix} \mathbf{C}^T & -\mathbf{A}^T\mathbf{C}^T & \left(-\mathbf{A}^T\right)^2\mathbf{C}^T & \dots & \left(-\mathbf{A}^T\right)^{n-1}\mathbf{C}^T \end{bmatrix}$ has full rank. Changing the sign of some of the columns does not alter the rank, nor does taking the transpose; therefore it follows that the following theorem is proved:

Theorem 6.25. *The control system* $\begin{matrix} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} \end{matrix}$ *in which* $\boldsymbol{A}$, $\boldsymbol{B}$, *and* $\boldsymbol{C}$ *are constant is completely observable if and only if the observability matrix*

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \boldsymbol{CA}^2 \\ \vdots \\ \boldsymbol{CA}^{n-1} \end{bmatrix},$$

which is a $np \times n$ *matrix, has rank* n.

As an example of this theory, we consider the motion example, where

$$\dot{\boldsymbol{x}} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\boldsymbol{x} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}u,$$
$$\boldsymbol{y} = x_1 \equiv \begin{pmatrix} 0 & 1 \end{pmatrix}\boldsymbol{x}.$$

Constructing the associated observability matrix $\mathbf{V}$, we have

$$\mathbf{V} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \Rightarrow rank\,(\mathbf{V}) = 2,$$

and therefore the control system is completely observable.

We now move on to an important feature of control theory which also plays an important role in numerical modeling of differential equations. This feature is referred to as **stability**.

6.6 Stability

Consider a general first-order system of ordinary differential equations for the vector $\boldsymbol{x} = \boldsymbol{x}(t)$,

$$\left.\begin{array}{l} \dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}, t) \\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0 \end{array}\right\} t > t_o, \tag{6.60}$$

we have the following definition for critical points.

Definition 6.26. A critical point, or point of equilibrium, is a point, $\boldsymbol{c}$, such that if $\boldsymbol{x}(t^*) = \boldsymbol{c}$ for some time $t^* > t_0$, then $\boldsymbol{x}(t) = \boldsymbol{c}$ for all t greater than t^*.

Note that for this definition to be true, we must have $\dot{\boldsymbol{x}}(t) = \mathbf{0}$ for $t > t_0$ and hence $\boldsymbol{f}(\boldsymbol{c}, t) = \mathbf{0}$ for $t > t^*$.

Without loss of generality it is possible to assume that $\boldsymbol{c} = \mathbf{0}$ is the point of equilibrium; however, if this is not the case then it is possible to shift the origin by putting $\hat{\boldsymbol{x}}(t) = \boldsymbol{x}(t) - \boldsymbol{c}$. Now, given an equilibrium point, there are three types of stability that could be associated with this point.

Definition 6.27. Suppose that $\boldsymbol{x} = \mathbf{0}$ is an equilibrium point. In that case:

1. This point is said to be **stable** if

$$\forall \varepsilon > 0 \, \exists \delta > 0 \quad \text{such that} \quad \left|\left|\boldsymbol{x}\left(t^*\right)\right|\right| < \delta \Rightarrow ||\boldsymbol{x}(t)|| < \varepsilon, \ \forall t > t^*.$$

2. This point is said to be **asymptotically stable** if it is **stable** and

$$||\boldsymbol{x}(t)|| \to 0 \text{ as } t \to \infty.$$

3. This point is said to be **unstable** if it is **not stable**, which implies

$$\exists \varepsilon > 0 \text{ such that } \forall \delta > 0 \, \exists \boldsymbol{x}\left(t^*\right) \text{ such that } \left|\left|\boldsymbol{x}\left(t^*\right)\right|\right| < \delta \text{ and } ||\boldsymbol{x}(t)|| \geq \varepsilon$$

for some $t > t^*$.

To illustrate, we present a simple descriptive example.

Example 6.28.

1. *A ball in a bowl in the absence of friction will oscillate forwards and backwards forever and with the same amplitude. Therefore this system is* ***stable****.*
2. *When friction is included the amplitudes of the oscillations will decrease (decay) with time and eventually the ball will come to rest at the bottom of the bowl. Therefore, this system is* ***asymptotically stable****.*
3. *If the same bowl was turned upside and the ball released at the top then the ball would roll off. Therefore, this system is* ***unstable****.*

We now consider linear time-invariant systems,

$$\left.\begin{array}{l} \dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \end{array}\right\} \quad t > 0, \tag{6.61}$$

where $\mathbf{A}$ is a constant matrix. Note that $\boldsymbol{x} = \mathbf{0}$ is an equilibrium point since if $\boldsymbol{x} = \mathbf{0}$ then $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} = \mathbf{0}$. If $\mathbf{A}$ is singular there may be other points of equilibrium, that is to say non-zero solutions to $\mathbf{A}\boldsymbol{x} = \mathbf{0}$.

The next step is to determine algebraic stability conditions to ensure in advance the stability of the control system.

6.6.1 Algebraic Stability Conditions

It is possible to classify equilibrium points according to the eigenvalues of the matrix $\mathbf{A}$. Suppose that the eigenvalues of $\mathbf{A}$ are denoted as λ_j for $j = 1, 2, \ldots, n$.

1. Suppose that $Re(\lambda_j) < 0$ for $j = 1, 2, \ldots, n$, as such each solution will be of the form $e^{\lambda_j t}\boldsymbol{v}_j$ if $\mathbf{A}$ has a full set of eigenvalues. If there is not a full set of eigenvectors for $\mathbf{A}$, then there will be solutions of the form $te^{\lambda_j t}$ and possible $t^2 e^{\lambda_j t}$. For all of these cases, all the solutions tend to zero as $t \to \infty$. Therefore, for this situation the control systems are **asymptotically stable**.
2. Suppose that $Re(\lambda_j) = 0$ for some j, and that for the remainder we have $Re(\lambda_j) < 0$. Therefore, the associated solutions for the eigenvalues with zero real component will be of the form $e^{(0+i\beta_j)t}\boldsymbol{v}_j$ or specifically $(\cos\beta_j + i\sin\beta_j t)\boldsymbol{v}_j$ which do not tend to zero at $t \to \infty$ but do remain bounded. Therefore, these points are **stable**. However, if λ_j does not have a full set of eigenvectors, the solutions will be of the form $t(\cos\beta_j + i\sin\beta_j t)\boldsymbol{v}_j$, which then grow unboundedly as $t \to \infty$ and are therefore **unstable**.
3. Suppose that $Re(\lambda_j) > 0$ for some j. Therefore the solutions are all growing exponentials which are not bounded and hence these points are **unstable**.

To summarize, we have shown that the equilibrium point $\boldsymbol{x} = \mathbf{0}$ is

1. asymptotically stable if and only if $Re(\lambda_j) < 0, \forall j = 1, 2, \ldots, n$; or
2. stable if and only if $Re(\lambda_j) \leq 0 \; \forall j = 1, 2, \ldots, n$ provided that any eigenvalue with a zero real part has a full set of eigenvectors associated with them.

To illustrate this further, we consider the following three examples for the problem $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$:

Example 6.29. $\mathbf{A} = \begin{pmatrix} 1 & -2 \\ -8 & 1 \end{pmatrix}$. *The associated eigenvalues are* $\lambda_1 = 5$ *and* $\lambda_2 = -3$. *Therefore, there are two solutions:* $x_1 = e^{5t}\boldsymbol{v}_1$ *which grows unboundedly and* $x_2 = e^{-3t}\boldsymbol{v}_2$ *which decays to zero. However, this implies that* $\boldsymbol{x} = \mathbf{0}$ *is an unstable equilibrium point.*

Example 6.30. $\mathbf{A} = \begin{pmatrix} -1 & 1 \\ -1 & -1 \end{pmatrix}$. *The associated eigenvalues for this matrix are* $\lambda_1 = -1 - i$ *and* $\lambda_2 = -1 + i$, *which have associated solutions of the form* $x_1 = e^{-t}(\cos -t + i\sin -t)\boldsymbol{v}_1$ *and* $x_2 = e^{-t}(\cos t + i\sin t)\boldsymbol{v}_2$. *Therefore, due to the negative exponential, both of the solutions tend to* $\boldsymbol{x} = \mathbf{0}$ *as* $t \to \infty$ *and as such* $\boldsymbol{x} = \mathbf{0}$ *is an asymptotically stable equilibrium point.*

Example 6.31. $\mathbf{A} = \begin{pmatrix} -1 & 1 \\ -5 & 1 \end{pmatrix}$. *The associated eigenvalues for this matrix are* $\lambda_1 = 2i$ *and* $\lambda_2 = -2i$ *which have associated solutions of the form* $x_1 = (\cos 2t + i\sin 2t)\boldsymbol{v}_1$ *and* $x_2 = (\cos -2t + i\sin -2t)\boldsymbol{v}_2$. *Therefore, these solutions neither decay nor grow unboundedly, but rather they oscillate. This then implies that* $\boldsymbol{x} = \mathbf{0}$ *is a stable equilibrium point.*

What these three examples are indicating is that the position of the eigenvalues on the complex plane is what governs stability. Eigenvalues are found by solving for the roots of the polynomial

$$\lambda^n + \gamma_{n-1}\lambda^{n-1} + \cdots + \gamma_2\lambda^2 + \gamma_1\lambda + \gamma_0 = 0. \tag{6.62}$$

We now examine such polynomials to determine when the eigenvalues are in certain parts of the complex plane.

Theorem 6.32. ***The Routh-Hurwitz Criterion****: Consider the polynomial of degree n,* $p(\lambda) = \sum_{j=1}^{n} \gamma_j \lambda^j$*, where* $\gamma_j \in \mathbb{R}$*, in which, without loss of generality, we can assume* $\gamma_n = 1$*, then*

$$Re(\lambda_k) < 0 \Rightarrow \gamma_j > 0 \quad (j = 0, 1, \ldots, n-1), \tag{6.63}$$

where λ_k *are the roots of* $p(\lambda)$*.*

It is possible to state two corollaries for the cases where $n = 2$ and $n = 3$.

Corollary 6.33. *If* $n = 2$ *then the Routh-Hurwitz criterion gives an equivalent condition:*

$$Re(\lambda_k) < 0 \Leftrightarrow \gamma_{0,1} > 0.$$

Corollary 6.34. *If* $n = 3$ *then we have the two conditions:*

$$\left.\begin{array}{l} \gamma_0, \gamma_1, \gamma_2 > 0 \\ \gamma_2\gamma_1 - \gamma_0 > 0 \end{array}\right\} \Leftrightarrow Re(\lambda_k) < 0, \quad k = 1, 2, 3.$$

To illustrate Corollary 6.34, we have the following general example.

Example 6.35. *Choose values for* f_1 *and* f_2 *in the polynomial*

$$p(\lambda) = \lambda^3 + f_1\lambda^2 + (f_1 + f_2)\lambda + f_1,$$

so as to force all of the roots of $p(\lambda)$ *into the left-hand half of the complex plane.*

The roots to the third-order polynomial above will only have positive real parts if and only if all three of the following equalities are satisfied:

$$\begin{array}{ll} f_1 > 0 & \text{(i)}, \\ f_1 + f_2 > 0 & \text{(ii)}, \\ f_1(f_1 + f_2) - f_1 > 0 & \text{(iii)}. \end{array} \tag{6.64}$$

It is possible to rearrange condition (iii) in (6.64) to be $f_1(f_1 + f_2 - 1) > 0$, which leads to the following two requirements:

$$\begin{array}{r} f_1 > 0, \\ f_1 + f_2 > 1, \end{array}$$

and therefore any choices of f_1 and f_2 satisfying the conditions above will be adequate. For example, if set $f_1 = 6$ and $f_2 = 5$, then we obtain roots $\lambda_1 = -1$, $\lambda_2 = -2$, and $\lambda_3 = -3$.

Exercise 6.36. *A second-order process is governed by the system of equations*

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & 1 \\ -b & -1 \end{pmatrix}\mathbf{x} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}u, \quad y = \begin{pmatrix} 0 & 1 \end{pmatrix}\mathbf{x}.$$

For what values of b is the system above (a) completely controllable; (b) completely observable; (c) stable near the equilibrium point $\mathbf{x} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$*; (d) asymptotically stable near the equilibrium point* $\mathbf{x} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$*?*

Exercise 6.37. *Determine the range of values for m such that the following uncontrolled system is asymptotically stable:*

$$\dot{\boldsymbol{x}} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -m & -1 & -2 \end{pmatrix} \boldsymbol{x}.$$

6.7 Feedback

The starting point for the derivation of feedback is to consider the control system

$$\begin{aligned} \dot{\boldsymbol{x}} &= \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u} \quad (t > t_0) \\ \boldsymbol{x}(t_0) &= \boldsymbol{x}_0, \\ \boldsymbol{y} &= \mathbf{C}\boldsymbol{x}. \end{aligned}$$

The control systems that we have considered so far are referred to as **open loop** control systems, where given a control, $\boldsymbol{u}$, we produce an output, $\boldsymbol{y}$.

Now we seek an automatic procedure for choosing $\boldsymbol{u}$, i.e., we monitor the output $\boldsymbol{y}$ and choose $\boldsymbol{u}$ accordingly. The first approach for achieving this is to express the control in terms of the output, for example, as

$$\boldsymbol{u} = \mathbf{F}\boldsymbol{y} + \boldsymbol{v}, \tag{6.65}$$

where $\mathbf{F} \in \mathbb{R}^{m \times p}$, and $\boldsymbol{v}$ is known as a reference vector and allows the user to retain some control. This type of control system is referred to as a **closed loop** control system. The mathematical expression for a closed loop control system is

$$\begin{aligned} \dot{\boldsymbol{x}} &= \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \\ &= \mathbf{A}\boldsymbol{x} + \mathbf{B}(\mathbf{F}\boldsymbol{y} + \boldsymbol{v}), \\ &= \mathbf{A}\boldsymbol{x} + \mathbf{B}(\mathbf{F}\mathbf{C}\boldsymbol{x} + \boldsymbol{v}), \\ &= (\mathbf{A} + \mathbf{B}\mathbf{F}\mathbf{C})\boldsymbol{x} + \mathbf{B}\boldsymbol{v}, \qquad (6.66) \\ &= \mathbf{A}_c\boldsymbol{x} + \mathbf{B}\boldsymbol{v}, \qquad (6.67) \end{aligned}$$

where the subscript refers to the *closed* loop.

There are two significant cases:

1. $\mathbf{C}$ has rank n, and without loss of generality $\mathbf{C} = \mathbf{I}$ and all of the states $\boldsymbol{x}$ are known. This is then referred to as **state feedback**. This makes all of the eigenvalues of $\mathbf{A}_c$ assignable as a result of the completely controllable property.
2. If the rank of $\mathbf{C}$ is less than the number of states, then we have what is referred to as **output feedback**. However, in this situation the eigenvalues of $\mathbf{A}_c$ may not be available for us to control.

Example 6.38. *This is an example of how to form a state feedback to make a control system's equilibrium point stable that is currently unstable. To help illustrate the structure of a state feedback, we have drawn a schematic of this setup in Fig. 6.1. We now consider how this would work with the*

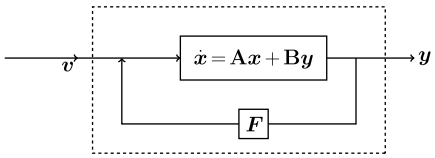


FIGURE 6.1

Schematic of a closed loop system.

motion example again to form a state feedback to make a control system's equilibrium point stable that is currently unstable. We start with

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u.$$

The reason why the equilibrium point is unstable is because we have only one eigenvalue for $\mathbf{A}$*, which is* $\lambda = 0$*; there is also only 1 eigenvector associated with that eigenvalue. We shall now use a state feedback to attain asymptotic stability.*

The first step is to create a control, u, of the form $u = \boldsymbol{F}\boldsymbol{x} + v$*, which results in*

$$u = \mathbf{F}\boldsymbol{x} + v = \begin{pmatrix} f_1 & f_2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + v,$$

$$\mathbf{A} + \mathbf{BF} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} f_1 & f_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ f_1 & f_2 \end{pmatrix}, \tag{6.68}$$

$$= \mathbf{A}_c. \tag{6.69}$$

As an easy illustration we shall choose to assign our new eigenvalues as $\lambda_1 = -1$ *and* $\lambda_2 = -3$*. Therefore, for it to be possible to have these two eigenvalues, we form*

$$det\,(\lambda\mathbf{I} - \mathbf{A}_c) = \begin{vmatrix} \lambda & -1 \\ -f_1 & \lambda - f_2 \end{vmatrix},$$

$$= \lambda^2 - \lambda f_2 - f_1 \equiv (\lambda + 1)(\lambda + 3).$$

Thus to make the two expressions above equivalent we need to set $f_1 = -3$ *and* $f_2 = -4$*. This then makes the new closed loop control system*

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -3x_1 - 4x_2 + v. \end{aligned} \tag{6.70}$$

However, the question remains: is it possible to assign eigenvalues when all that is available is output feedback?

Example 6.39. *We again consider the same motion example control system, but now we have observations/outputs from the control system. Mathematically this is expressed as*

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u,$$

$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = x_1.$$

From our previous work with this example we have shown that this control system is both completely controllable and completely observable. We shall now attempt to make this system asymptotically stable by trying the output feedback control as $u = f\boldsymbol{y} + \boldsymbol{v}$*:*

$$\begin{aligned} \mathbf{A} + \mathbf{BFC} &= \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} f \begin{pmatrix} 1 & 0 \end{pmatrix}, \\ &= \begin{pmatrix} 0 & 1 \\ f & 0 \end{pmatrix} \equiv \mathbf{A}_c. \end{aligned}$$

Finding the eigenvalues of $\mathbf{A}_c$ *is accomplished through*

$$|\lambda \mathbf{I} - \mathbf{A}_c| = \begin{vmatrix} \lambda & -1 \\ -f & \lambda \end{vmatrix} = \lambda^2 - f = 0.$$

There are three possible sets of values that f *can take. It could be greater than zero, equal to zero, or less than zero. We now consider outcomes from these three choices:*

- $f > 0 \Rightarrow$ *The associated equilibrium point will be* ***unstable*** *as one of the eigenvalues will be positive.*
- $f < 0 \Rightarrow$ *There would be two imaginary eigenvalues* $\pm\sqrt{-f}\,i$*. Therefore, the associated equilibrium point would be* ***stable*** *but it would* ***not be asymptotically stable****.*
- $f = 0 \Rightarrow$ *Under this scenario there would be a repeated eigenvalue with only one associated eigenvector and therefore the equilibrium point would remain* ***unstable****.*

As we can see from the example above, if we have a control that is of the form of output feedback, then it may not always be possible to make the equilibrium point asymptotically stable. We have seen that this is possible through state feedback, however, that approach may not always be possible. We now consider new estimators that are a combination of *observers* and *state* estimators.

6.7.1 Observers and State Estimators

The motivation for this subsection is to seek an approximation to $\boldsymbol{x}$ which could be used in a form of state feedback. The aim is to design a control system whose output is an approximation to $\boldsymbol{x}$.

We are going to design a control system that is of the form

$$\dot{\hat{\boldsymbol{x}}} = \mathbf{A}\hat{\boldsymbol{x}} + \mathbf{B}\boldsymbol{u} + \mathbf{G}\left(\mathbf{C}\hat{\boldsymbol{x}} - \boldsymbol{y}\right), \tag{6.71}$$

where $\boldsymbol{u}$ is the input to the original control system, and $\boldsymbol{y}$ is an output of the original system. We need to ensure that $\left(\hat{\boldsymbol{x}} - \boldsymbol{x}\right) \to 0$ as $t \to \infty$, and so we need to choose $\mathbf{G}$ to make it happen. We have provided a schematic of this form of feedback in Fig. 6.2.

To be able to choose $\mathbf{G}$ we require the *error* between the original control systems' solution and the solution to the control system in (6.71), denoted as $\boldsymbol{e}$, to tend to zero; that is to say we require $\boldsymbol{e} \equiv \hat{\boldsymbol{x}} - \boldsymbol{x} \to \boldsymbol{0}$. Now, as just mentioned, we know that the associated control system is given by

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \tag{6.72}$$

$$\dot{\hat{\boldsymbol{x}}} = \mathbf{A}\hat{\boldsymbol{x}} + \mathbf{B}\boldsymbol{u} + \mathbf{G}\left(\mathbf{C}\hat{\boldsymbol{x}} - \boldsymbol{y}\right). \tag{6.73}$$

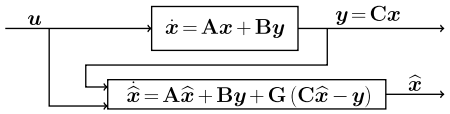


FIGURE 6.2

Schematic of the observer/state estimators system control system.

Taking the difference between (6.73) and (6.72) results in an expression for the time evolution of the error, that is to say

$$\begin{aligned}\dot{\boldsymbol{e}} &\equiv \dot{\hat{\boldsymbol{x}}} - \dot{\boldsymbol{x}},\\ &= \mathbf{A}\boldsymbol{e} + \mathbf{GC}\hat{\boldsymbol{x}} - \mathbf{G}\boldsymbol{y}, \qquad (6.74)\\ &= (\mathbf{A} + \mathbf{GC})\,\boldsymbol{e}, \qquad (6.75)\end{aligned}$$

where we have substituted $\mathbf{C}\boldsymbol{x}$ for $\boldsymbol{y}$ in (6.74) to obtain the final expression in (6.75). Therefore, we can only have $\boldsymbol{e} \to 0$ as $t \to \infty \Leftrightarrow$ the eigenvalues of $\mathbf{A} + \mathbf{GC}$ have negative real part.

Now if $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$, $\boldsymbol{y} = \mathbf{C}\boldsymbol{x}$ is completely observable, then the dual control system $\dot{\boldsymbol{x}} = -\mathbf{A}^T\boldsymbol{x} + \mathbf{C}^T\boldsymbol{u}$ is completely controllable and therefore by previously established results we have that we can use $\mathbf{G}$ to assign any eigenvalues to $-\mathbf{A}^T - \mathbf{C}^T\mathbf{G}^T$, which has the same eigenvalues as $-(\mathbf{A} + \mathbf{GC})^T$ whose eigenvalues are the minuses of those of $\mathbf{A} + \mathbf{GC}$.

Therefore, in summary: if the system $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$, $\boldsymbol{y} = \mathbf{C}\boldsymbol{x}$ is completely observable, then it is always possible to choose $\mathbf{G}$ in the control system

$$\dot{\hat{\boldsymbol{x}}} = \mathbf{A}\hat{\boldsymbol{x}} + \mathbf{B}\boldsymbol{u} + \mathbf{G}\left(\mathbf{C}\hat{\boldsymbol{x}} - \boldsymbol{y}\right),$$

so that $\hat{\boldsymbol{x}} - \boldsymbol{x} \to \mathbf{0}$ as $t \to 0$.

We return to the motion example, and now consider building an observer to make the equilibrium point asymptotically stable.

Example 6.40. *As before, we have that*

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}0\\ 1\end{pmatrix}u,$$
$$y = \begin{pmatrix}1 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix}.$$

We have already shown that this control system is completely controllable and completely observable. Therefore, we are going to build an observer of the form

$$\mathbf{A} + \mathbf{GC} = \begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix} + \begin{pmatrix}g_1\\ g_2\end{pmatrix} = \begin{pmatrix}g_1 & 1\\ g_2 & 0\end{pmatrix}. \qquad (6.76)$$

We now require the eigenvalues of the matrix in (6.76) to have negative real parts, therefore if we choose to have $\lambda_1 = -1$ and $\lambda_2 = -2$, then we have

$$\begin{aligned}det\,(\lambda\mathbf{I} - (\mathbf{A} + \mathbf{GC})) &= \begin{vmatrix}\lambda - g_1 & 1\\ g_2 & \lambda\end{vmatrix} = \lambda^2 - g_1\lambda - g_2,\\ &= (\lambda + 1)(\lambda + 2).\end{aligned}$$

To make the two expressions above equivalent, we require $g_1 = -3$ and $g_2 = -2$. Therefore, it is possible to choose g_1 and g_2 such that we obtained our desired eigenvalues. Hence the associated observer system is

$$\dot{\hat{\boldsymbol{x}}} = \begin{pmatrix} -3 & 1 \\ -2 & 0 \end{pmatrix} \hat{\boldsymbol{x}} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u - \begin{pmatrix} -3 \\ -2 \end{pmatrix} \boldsymbol{y}, \tag{6.77}$$

and has the property that $\left(\hat{\boldsymbol{x}} - \boldsymbol{x}\right) \to \mathbf{0}$ as $t \to \infty$.

The next step is to use $\hat{\boldsymbol{x}}$ in a feedback. Therefore, we are going to create a control of the form

$$\boldsymbol{u} = \mathbf{F}\hat{\boldsymbol{x}} + \boldsymbol{v}. \tag{6.78}$$

Given the control in (6.78), we finally have to show that for a system that is completely controllable and completely observable, then it is possible to choose $\mathbf{F}$ and $\mathbf{G}$ so that the **joint** control system

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}, \tag{6.79}$$

$$\dot{\hat{\boldsymbol{x}}} = \mathbf{A}\hat{\boldsymbol{x}} + \mathbf{B}\boldsymbol{u} + \mathbf{G}\left(\mathbf{C}\hat{\boldsymbol{x}} - \boldsymbol{y}\right), \tag{6.80}$$

or

$$\begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\hat{\boldsymbol{x}}} \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{GC} & \mathbf{A} + \mathbf{GC} \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{B} \end{pmatrix} \boldsymbol{u}, \tag{6.81}$$

is asymptotically stable. Applying the feedback $\boldsymbol{u} = \mathbf{F}\hat{\boldsymbol{x}} + \boldsymbol{v}$, we obtain the closed loop control system

$$\begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\hat{\boldsymbol{x}}} \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{BF} \\ -\mathbf{GC} & \mathbf{A} + \mathbf{GC} + \mathbf{BF} \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{x}} \end{pmatrix} + \begin{pmatrix} \mathbf{B} \\ \mathbf{B} \end{pmatrix} \boldsymbol{u}. \tag{6.82}$$

To achieve asymptotic stability, we now move our attention to the eigenvalues of

$$\mathbf{A}_c = \begin{pmatrix} \mathbf{A} & \mathbf{BF} \\ -\mathbf{GC} & \mathbf{A} + \mathbf{GC} + \mathbf{BF} \end{pmatrix}.$$

Before we arrive at the desired result, we consider the following lemmas and theorems.

Lemma 6.41. *If $\boldsymbol{P}$ is a non-singular matrix then $\boldsymbol{A}$ has the same eigenvalues as $\boldsymbol{PAP}^{-1}$.*

Proof. Suppose that λ is an eigenvalue of $\mathbf{A}$, then

$$\begin{aligned} |\mathbf{A} - \lambda\mathbf{I}| = 0 &\Leftrightarrow |\mathbf{P}|\,|\mathbf{A} - \lambda\mathbf{I}|\left|\mathbf{P}^{-1}\right| = 0, \\ &\Leftrightarrow \left|\mathbf{PAP}^{-1} - \lambda\mathbf{PP}^{-1}\right| = 0, \\ &\Leftrightarrow \left|\mathbf{PAP}^{-1} - \lambda\mathbf{I}\right| = 0, \end{aligned}$$

and hence λ is an eigenvalue of $\mathbf{PAP}^{-1}$ and therefore the lemma is proven.

This leads into the following important theorem:

Theorem 6.42. *If the control system* $\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu}$, $\boldsymbol{y} = \boldsymbol{Cx}$ *is completely controllable and completely observable, then it is possible to choose* $\boldsymbol{F}$ *and* $\boldsymbol{G}$ *in*

$$\boldsymbol{A}_c = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{BF} \\ -\boldsymbol{GC} & \boldsymbol{A} + \boldsymbol{GC} + \boldsymbol{BF} \end{pmatrix}, \tag{6.83}$$

so that $\boldsymbol{A}_c$ *has any prescribed set (closed under conjugation) of eigenvalues.*

Proof. Let $\mathbf{P} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & -\mathbf{I} \end{pmatrix}$, which has the property that $\mathbf{P}^2 = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$ which indicates that $\mathbf{P}$ is its own inverse $\mathbf{P}^{-1}$. It follows from the lemma above that $\mathbf{PA}_c\mathbf{P}$ has the same eigenvalues as $\mathbf{A}_c$. Therefore, forming $\mathbf{PA}_c\mathbf{P}$ we have

$$\begin{aligned} \mathbf{PA}_c\mathbf{P} &= \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & -\mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{A} & \mathbf{BF} \\ -\mathbf{GC} & \mathbf{A} + \mathbf{GC} + \mathbf{BF} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & -\mathbf{I} \end{pmatrix}, \\ &= \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I} & -\mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{A} + \mathbf{BF} & -\mathbf{BF} \\ \mathbf{A} + \mathbf{BF} & -\mathbf{A} - \mathbf{GC} - \mathbf{BF} \end{pmatrix}, \\ &= \begin{pmatrix} \mathbf{A} + \mathbf{BF} & -\mathbf{BF} \\ \mathbf{0} & \mathbf{A} + \mathbf{GC} \end{pmatrix}, \end{aligned} \tag{6.84}$$

which has the eigenvalues of $\mathbf{A} + \mathbf{BF}$ and those of $\mathbf{A} + \mathbf{GC}$. We know that the original control system is completely controllable and therefore we can choose $\mathbf{F}$ so that $\mathbf{A} + \mathbf{BF}$ has specific eigenvalues. We have that the original control system was also completely observable and so it is possible to choose $\mathbf{G}$ so that $\mathbf{A} + \mathbf{GC}$ has specified eigenvalues. Therefore it is possible to choose a $\mathbf{F}$ and a $\mathbf{G}$ so that $\mathbf{A}_c$ has any prescribed set of eigenvalues. We have provided a schematic of the full feedback system in Fig. 6.3.

Example 6.43. *We return again to the motion example where we have the control system given by*

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u,$$
$$y = x_1.$$

It has been shown in earlier examples that is possible to designed an observer with $\mathbf{G} = \begin{pmatrix} -3 \\ -2 \end{pmatrix}$ *so that* $\mathbf{A} + \mathbf{GC} = \begin{pmatrix} -3 & 1 \\ -2 & 0 \end{pmatrix}$ *has eigenvalues* $\lambda_1 = -1$ *and* $\lambda_2 = -2$. *In another earlier example it was shown that it is also possible to designed a state feedback with* $\mathbf{F} = \begin{pmatrix} -3 & -4 \end{pmatrix}$, *therefore making*

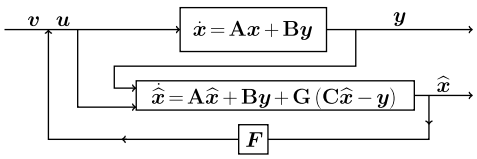


FIGURE 6.3

Schematic of the full feedback control system.

$\mathbf{A}+\mathbf{BF}=\begin{pmatrix}0 & 1\\ -3 & -4\end{pmatrix}$ *which then had eigenvalues* $\lambda_1=-1$ *and* $\lambda_2=-3$. *Given these expressions, it is now possible to define a closed loop, joint system for the motion example as*

$$\begin{aligned}\mathbf{A}_c&=\begin{pmatrix}\mathbf{A} & \mathbf{BF}\\ -\mathbf{GC} & \mathbf{A}+\mathbf{GC}+\mathbf{BF}\end{pmatrix},\\ &=\begin{pmatrix}0 & 1 & 0 & 0\\ 0 & 0 & -3 & -4\\ 3 & 0 & -3 & 1\\ 2 & 0 & -5 & -4\end{pmatrix}.\end{aligned}\tag{6.85}$$

Therefore, by the theorem above, the joint control system has eigenvalues $\lambda_1=-1$, $\lambda_2=-1$, $\lambda_3=-2$, *and* $\lambda_4=-3$.

In summary for the motion example, it has been shown that given a control system of the form

$$\begin{aligned}\dot{\boldsymbol{x}}&=\mathbf{A}\boldsymbol{x}+\mathbf{B}\boldsymbol{u},\\ \boldsymbol{y}&=\mathbf{C}\boldsymbol{x}\end{aligned}$$

for $n=2$, which is completely controllable and completely observable but which cannot be made asymptotically stable through output feedback, then the larger problem

$$\begin{pmatrix}\dot{\boldsymbol{x}}\\ \dot{\hat{\boldsymbol{x}}}\end{pmatrix}=\mathbf{A}_c\begin{pmatrix}\boldsymbol{x}\\ \hat{\boldsymbol{x}}\end{pmatrix}+\begin{pmatrix}\mathbf{B}\\ \mathbf{B}\end{pmatrix}\boldsymbol{u}$$

with output $\begin{pmatrix}\boldsymbol{y}\\ \hat{\boldsymbol{x}}\end{pmatrix}$ is asymptotically stable at the equilibrium point.

6.8 Summary

In this chapter we have introduced the theory for solving control problems. We have introduced the state transition matrix, as well as the conditions for completely controllability, through the controllability matrix, completely observable, through the observability matrix, as well as duality linking the controllability and observability properties in either observation space or control space. This duality property is important later on in the derivation of two different forms of variational data assimilation. We have also introduced the properties of stability, asymptotic stability, and instability of control problems to understand if the solution to the control problem is bounded, decaying, or growing. Finally, we introduced different forms of feedback to enable the control problem to have specific stability properties either through observability feedback or through a control feedback. Feedback is an important property in the derivation of the Kalman filter, which we shall go into more detail about later.

Given these properties of the control problem, we now extend this theory, along with using and extending some of the results from the calculus of variation chapter, to find the **optimal control**.

CHAPTER

Optimal Control Theory

7

Contents

The sets of problem examined in the previous chapter were associated with how to choose a control that achieves some desired result. An example of this would be to move a control system $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ at some initial time to some final state $\boldsymbol{x}(t_f)$ at some final time t_f, where $t_f > t_0$. However, in practice there could be extra constraints on the control, i.e., the most efficient, the shortest time, the quickest acceleration, and therefore we require techniques to find this most efficient controls, which is more commonly referred to as the finding or solving for the **optimal controls**.

It was shown in Section 6.3 that the control

$$\boldsymbol{u}^*(t) = -\mathbf{B}^T(t)\,\boldsymbol{\Phi}(t_0,t)^T\,\mathbf{M}^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{\Phi}(t_0,t_f)\boldsymbol{x}_f\right), \tag{7.1}$$

moves the state $\boldsymbol{x}$, which is the solution to $\dot{\boldsymbol{x}}(t) = \mathbf{A}(t)\boldsymbol{x}(t) + \mathbf{B}(t)\boldsymbol{u}(t)$ from $\boldsymbol{x}_0$ at time t_0 to $\boldsymbol{x}_f$ at time t_f. The controllability matrix, $\mathbf{M}$, for the time variate case is defined as

$$\mathbf{M} = \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0,s)\,\mathbf{B}(s)\,\mathbf{B}^T(s)\,\boldsymbol{\Phi}(t_0,t)^T\,ds,$$

and is required to be non-singular for the validity of the control defined in (7.1).

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00011-5

The control $\boldsymbol{u}^*(t)$ defined in (7.1) is determined to be **optimal** in the sense that the integral

$$D\left[\boldsymbol{u}^*(t)\right] = \int_{t_0}^{t_f} \boldsymbol{u}^*(s)^T \boldsymbol{u}^*(s)\, ds, \tag{7.2}$$

is minimized. Therefore, all other controls $\boldsymbol{u}(t)$ with the property $\boldsymbol{x}\left(t_f\right) = \boldsymbol{x}_f$ are such that $D[\boldsymbol{u}(t)] > D\left[\boldsymbol{u}^*(t)\right]$.

We shall now prove that this is true for a control system situation; however, the formulation above may appear familiar for those who have worked through the calculus of variation chapter.

Proof. By the assumption made above, we have that both controls achieve the required objective that $\boldsymbol{x}\left(t_f\right) = \boldsymbol{x}_f$. Therefore, we have that

$$\boldsymbol{x}_f = \boldsymbol{\Phi}\left(t_f, t_0\right)\left\{\boldsymbol{x}_0 + \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0, s)\,\mathbf{B}(s)\,\boldsymbol{u}(s)\, ds\right\}, \tag{7.3}$$

and

$$\boldsymbol{x}_f = \boldsymbol{\Phi}\left(t_f, t_0\right)\left\{\boldsymbol{x}_0 + \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0, s)\,\mathbf{B}(s)\,\boldsymbol{u}^*(s)\, ds\right\}. \tag{7.4}$$

Subtracting (7.4) from (7.3) and applying the invertibility property of the state transition matrix results in

$$\mathbf{0} = \int_{t_0}^{t_f} \boldsymbol{\Phi}(t_0, s)\,\mathbf{B}(s)\left[\boldsymbol{u}(s) - \boldsymbol{u}^*(s)\right] ds. \tag{7.5}$$

Next, multiplying (7.5) by $\left(\boldsymbol{x}_0 - \boldsymbol{\Phi}\left(t_0, t_f\right)\boldsymbol{x}_f\right)\mathbf{M}^{-T}$ results in

$$\mathbf{0} = \int_{t_0}^{t_f} \boldsymbol{u}^{*T}(s)\left[\boldsymbol{u}(s) - \boldsymbol{u}^*(s)\right] ds \Rightarrow \int_{t_0}^{t_f} \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}(s)\, ds = \int_{t_0}^{t_f} \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}^*(s)\, ds. \tag{7.6}$$

The next step is to consider the integral

$$\int_{t_0}^{t_f} \left[\boldsymbol{u}(s) - \boldsymbol{u}^*(s)\right]^T \left[\boldsymbol{u}(s) - \boldsymbol{u}^*(s)\right] ds > 0, \tag{7.7}$$

where for the integral in (7.7) to be positive definite, it is implied that $\boldsymbol{u}(t) \neq \boldsymbol{u}^*(t)$. Expanding the product in (7.7) results in

$$\int_{t_0}^{t_f} \boldsymbol{u}^T(s)\,\boldsymbol{u}(s) - \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}(s) - \boldsymbol{u}^T(s)\,\boldsymbol{u}^{*T}(s) + \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}^*(s)\, ds > 0. \tag{7.8}$$

Substituting the result from (7.6) into (7.8) results in

$$\int_{t_0}^{t_f} \boldsymbol{u}^T(s)\,\boldsymbol{u}(s) - 2\boldsymbol{u}^{*T}(s)\,\boldsymbol{u}^*(s) + \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}^*(s)\, ds > 0.$$

After cancelations and rearrangement, we obtain

$$\int_{t_0}^{t_f} \boldsymbol{u}^T(s)\,\boldsymbol{u}(s)\, ds > \int_{t_0}^{t_f} \boldsymbol{u}^{*T}(s)\,\boldsymbol{u}^*(s)\, ds,$$

which proves the requirement for an optimal control stated earlier.

To solve this type of problem we shall need to use some calculus of variation techniques introduced in Chapter 5. An important feature to notice here is that the integral, and the associated inequality defined in (7.7), is of the form of a **least squares problem**, which is closely associated with many different forms of data assimilation.

Before we start the derivation of some of the optimal control theory, we introduce a couple of definitions for types of functions, that the optimal control and state can take, that were not considered in Chapter 5. These are as follows:

Definition 7.1. A **piecewise continuous function** is a function that is continuous at all but a finite number of points $t_k \in (t_0, t_f)$ $k = 1, 2, \ldots, S$. At these finite number of points there may be a finite **jump discontinuity**, implying that the limits from the right and from the left at these points exist but can be different.

It should be noted here that for the control problem we assumed that neither the initial time, t_0, nor the final time, t_f, are one of the points where a jump occurs. We have plotted some examples of continuous, piecewise and non-piecewise functions in Fig. 7.1. Another property of the piecewise continuous functions on the interval $[t_0, t_f]$ is that the function is also bounded on this interval. For the control problem we assume that the controls are piecewise continuous, where the piecewise continuous property is denoted by $D_0[t_0, t_f]$.

Definition 7.2. A **piecewise smooth function** $\boldsymbol{x}(t)$ is a function that is continuous $\forall t \in [t_0, t_f]$ and is differentiable at all but a finite number of points $t_k \in (t_0, t_f)$ $t_k = 1, 2, \ldots, S$, at which the derivative has a jump continuity. That is to say the derivatives are piecewise continuous.

Fig. 7.2 shows three different functions to illustrate examples that are continuous, piecewise smooth, and piecewise continuous. In the piecewise smooth example (Fig. 7.2B) we have three points, t_1, t_2, and t_3, where the function has jumps in the derivatives. These points are referred to as **corners**. It should also be noted that piecewise smooth functions result from integrating piecewise continuous functions. Finally, the set of piecewise smooth functions on the interval $[t_0, t_f]$ are denoted as $D_1[t_0, t_f]$.

7.1 Optimizing Scalar Control Problems

As we showed at the beginning of this chapter, optimal control theory is dependent on using techniques from calculus of variations. We start by considering the cost functional

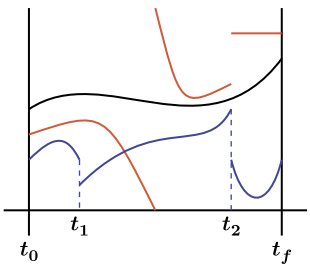


FIGURE 7.1

Examples of continuous (*black*), piecewise continuous (*blue*), and non-piecewise continuous functions (*red*).

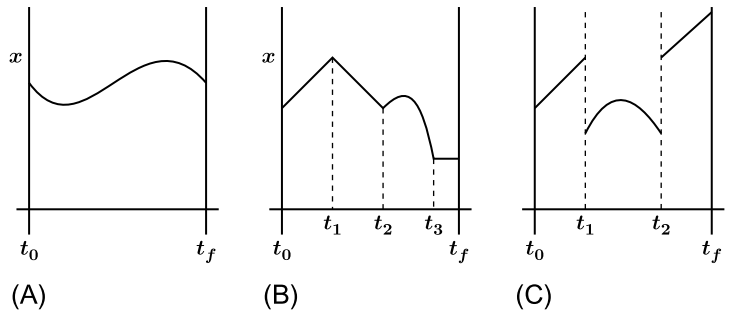


FIGURE 7.2

Examples of (A) continuous functions, (B) piecewise smooth function, and (C) piecewise continuous, but not piecewise smooth.

$$J(x) = \max_{u \in \mathcal{U}} \int_{t_0}^{t_f} F(x,u,t)\,dt,$$

subject to

$$\begin{aligned} \dot{x} &= f(x,u,t), \\ x(t_0) &= x_0, \\ x(t_f) &= x_f, \end{aligned} \tag{7.9}$$

where the expression $u \in \mathcal{U}$ refers to "all of the possible admissible controls" for this problem. An important feature of (7.9) is that while the control problem is an unconstrained problem, it describes a constrained optimization/calculus of variation problem.

The first step in solving (7.9) is to assume that x and u are independent variables that are constrained by the differential equation in (7.9); we therefore use the method of Lagrange multipliers, as shown in Chapter 5. Therefore, we rewrite (7.9) as

$$\max_{x,u} \mathcal{L}[x,u,\lambda] = \int_{t_0}^{t_f} F(x,u,t) + \lambda\,(f(x,u,t) - \dot{x})\,dt, \tag{7.10}$$

where it is assumed that the Lagrange multiplier, λ, is in the same set of functions as x. Thus the problem that has to be solved is to maximize $\mathcal{L}$ over all of the functions that satisfy the given constraint, and is equivalent to maximizing the original cost functional, J, subject to the constraint given.

We now introduce the function

$$G\left(x,u,\lambda,\dot{x},\dot{u},\dot{\lambda},t\right) = F(x,u,t) + \lambda\,(f(x,u,t) - \dot{x}), \tag{7.11}$$

and apply calculus of variation techniques, assuming that $x, u, \lambda \in \mathbb{C}_2\left[t_0, t_f\right]$. Therefore, as shown in Chapter 5, the next step is to form the system of Euler equations, which act as necessary conditions to find the optimal solution. We have three variables to differentiate the function in (7.11) with respect to x, u, and λ, which lead to

$$\frac{\partial G}{\partial x} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{x}}\right) = 0, \tag{7.12a}$$

$$\frac{\partial G}{\partial u} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{u}}\right) = 0, \tag{7.12b}$$

$$\frac{\partial G}{\partial \lambda} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\lambda}}\right) = 0. \tag{7.12c}$$

If $x\left(t_f\right)$ is not given, then from Chapter 5, we require a transversality condition, given by

$$\left.\frac{\partial G}{\partial \dot{x}}\right|_{t_f} = 0. \tag{7.13}$$

The system of differential equations in (7.12a)–(7.12c) can easily be verified to be

$$F_x + \lambda f_x + \frac{d}{dt}(\lambda) = 0, \tag{7.14a}$$

$$F_u + \lambda f_u = 0, \tag{7.14b}$$

$$f(x,u,t) - \dot{x} = 0, \tag{7.14c}$$

where (7.14c) is the original constraint.

We now introduce the **Hamiltonian**, which is denoted by $H(x,u,\lambda,t)$, and is defined for this problem as

$$H(x,u,\lambda,t) \equiv F(x,u,t) + \lambda f(x,u,t). \tag{7.15}$$

This now enables us to rewrite the functional as

$$G = F + \lambda(f - \dot{x}) \equiv H - \lambda\dot{x}, \tag{7.16}$$

which results in the following set of equations to solve for the optimal solution:

$$\dot{x} = f(x,u,t) \qquad \equiv \frac{\partial H}{\partial \lambda}, \tag{7.17a}$$

$$\dot{\lambda} = -(F_x + \lambda f_x) \quad \equiv -\frac{\partial H}{\partial x}, \tag{7.17b}$$

$$0 = F_u + \lambda f_u \qquad \equiv \frac{\partial H}{\partial u}, \tag{7.17c}$$

where (7.17a) is referred to as the **state equation**, (7.17b) is referred to as the **adjoint equation**, and finally (7.17c) is referred to as the **maximum principle**.

To help illustrate the principles presented above, we introduce a new example, which is based upon fisheries.

Example 7.3. ***Fisheries:*** *The state variable, x, is the fish population; the control is to extract the fish, $f(x)$ is the natural rate of growth. The associated state equation is $\dot{x} = f(x) - u$, $x(0) = x_0$ is the initial number of fish, $x(T) = x_T$ is the desired fish at time T. Two examples of possible models for the rate of growth of the fish are*

$$f(x) = rx\left(1 - \frac{x}{k}\right), \qquad f(x) = rx\ln\frac{k}{x}. \tag{7.18}$$

We now introduce a new function, $V(u)$, which is the value of the fish extracted; $C(x,u)$ is the cost of catching the fish and $\Pi(x,u) \equiv V(u) - C(x,u)$ is the profit. Therefore the problem becomes to maximize the objective functional

$$\max \int_0^T e^{-\delta t} \Pi(x,u)\, dt, \quad 0 < \delta < 1, \tag{7.19}$$

where δ is the discount factor.

A simplification of the fish problem is to assume that the rate of growth is $f(x) = x$, $\delta = 0$ and $\Pi = K - (u - u^*)^2$, where K is the cost of catching u^* fish. This then makes the functional

$$\begin{aligned} \max \int_0^T e^{-\delta t} \Pi dt &= \max \int_0^T K - \left(u - u^*\right)^2 dt, \\ &= \min \int_0^T \left(u - u^*\right)^2 dt, \end{aligned} \tag{7.20}$$

subject to $\dot{x} = x - u$, $x(0) = x_0$, $x(T) = x_T$.

If we consider the case where $T = 1$ and let $u^* = 0$, then we have the problem $\min \int_0^1 u^2\, dt$, subject to $\dot{x} = x - u$ with $x(0) = 0$ and $x(1) = x_1$.

The associated Hamiltonian is $H = -u^2 + \lambda(x - u - \dot{x})$, where the functional problem becomes

$$\max \mathcal{L} = \int_0^1 -u^2 + \lambda(x - u - \dot{x})\, dt, \tag{7.21}$$

and the associated system of Euler equations for (7.21) are

$$\dot{x} = x - u, \tag{7.22a}$$

$$\dot{\lambda} = -\frac{\partial H}{\partial x} = -\lambda \Rightarrow \lambda = ke^{-t}, k \in \mathbb{R}, \tag{7.22b}$$

$$0 = \frac{\partial H}{\partial u} = -2u - \lambda \Rightarrow u = -\frac{\lambda}{2} = -\frac{k}{2}e^{-t}. \tag{7.22c}$$

Substituting (7.22c) into (7.22a) results in

$$\dot{x} = x - u \equiv x + \frac{k}{2}e^{-t}, \tag{7.23}$$

where by applying the theory of integrating factors, we can show that x is equivalent to

$$x = Ce^t - \frac{k}{4}e^{-t}, \quad C \in \mathbb{R}, \quad k \in \mathbb{R}.$$

Applying the boundary conditions enables expressions to be found for the two constants C and k as

$$C = \frac{x_1 - x_0 e^{-1}}{2\sinh 1} \qquad -\frac{k}{4} = \frac{x_0 e^1 - x_1}{2\sinh 1}.$$

Thus the optimal control and the solution is defined as

$$\begin{aligned} u &= \frac{x_0 e - x_1}{\sinh 1} e^{-t}, \\ x &= \frac{1}{2\sinh 1}\left[\left(x_1 - x_0 e^{-1}\right)e^t + \left(x_0 e^1 - x_1\right)e^{-t}\right]. \end{aligned}$$

7.2 Multivariate Case

In many more advanced applications, it is possible that the problem has multiple state variables and controls and we may wish to derive a set of conditions to find the optimal control for the problem. The starting point is to consider the functional

$$\max \mathcal{L}[\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t] = \int_{t_0}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T (\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) - \dot{\boldsymbol{x}})\, dt, \tag{7.24}$$

where

$$\boldsymbol{\lambda}^T (\boldsymbol{f} - \dot{\boldsymbol{x}}) \equiv \sum_{i=1}^{n} \lambda_i (f_i - \dot{x}_i),$$

and we assume $\{x_i, u_i, \lambda_i\} \in \mathbb{C}_2 [t_0, t_f]$.

The next step is to form the function

$$G = F(x_1, \ldots, x_x, u_1, \ldots, u_m, t) + \sum_{s=1}^{n} \lambda_s (f_s(x_1, \ldots, x_n, u_1, \ldots, u_m, t) - \dot{x}_s). \tag{7.25}$$

Applying calculus of variation techniques to (7.25) results in the following systems of Euler equations:

$$\frac{\partial G}{\partial x_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{x}_i}\right) = 0, \quad i = 1, 2, \ldots, n, \tag{7.26a}$$

$$\frac{\partial G}{\partial u_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{u}_i}\right) = 0, \quad i = 1, 2, \ldots, m, \tag{7.26b}$$

$$\frac{\partial G}{\partial \lambda_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\lambda}_i}\right) = 0, \quad i = 1, 2, \ldots, n. \tag{7.26c}$$

If the boundary conditions, $x_i(t_f)$, are not given, then the associated transversality conditions are

$$\left.\frac{\partial G}{\partial x_i}\right|_{t_f} = 0.$$

Rewriting (7.26a)–(7.26c) in terms of the original functional results in

$$f_i - \dot{x}_i = 0, \qquad i = 1, 2, \ldots, n, \tag{7.27a}$$

$$\frac{\partial F}{\partial x_i} + \sum_{s=1}^{n} \lambda_s \frac{\partial f_s}{\partial x_i} + \frac{d}{dt} = 0, \qquad i = 1, 2, \ldots, n, \tag{7.27b}$$

$$\frac{\partial F}{\partial u_i} + \sum_{s=1}^{n} \lambda_s \frac{\partial f_s}{\partial u_i} = 0 \qquad i = 1, 2, \ldots, m, \tag{7.27c}$$

$$\lambda_i(t_f) = 0, \qquad x(t_f) = ?$$

The next step is to introduce the multivariate version of the Hamiltonian, which is given by

$$H = F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t). \tag{7.28}$$

We introduce the following notation to express the derivative of the Hamiltonian with respect to a vector as

$$\frac{\partial H}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial H}{\partial x_1} \\ \frac{\partial H}{\partial x_2} \\ \vdots \\ \frac{\partial H}{\partial x_n} \end{pmatrix}, \qquad \frac{\partial H}{\partial \boldsymbol{u}} = \begin{pmatrix} \frac{\partial H}{\partial u_1} \\ \frac{\partial H}{\partial u_2} \\ \vdots \\ \frac{\partial H}{\partial u_m} \end{pmatrix}, \tag{7.29}$$

which then enables us to define the multivariate version of the state, adjoint, and maximum principle equations as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) \equiv \frac{\partial H}{\partial \boldsymbol{\lambda}},$$

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \boldsymbol{x}}, \tag{7.30}$$

$$\boldsymbol{0} = \frac{\partial H}{\partial \boldsymbol{u}}. \tag{7.31}$$

Example 7.4. *Consider a mass on a spring with a force applied where we have the two states x_1 as the position and x_2 as the velocity. We are seeking the control u that minimizes the functional*

$$\int_0^{\frac{\pi}{2}} u^2 dt,$$

with the state equations

$$\dot{x}_1 = x_2,$$
$$\dot{x}_2 = -x_1 + u,$$

subject to the boundary conditions

$$x_1(0) = 0, \quad x_1\left(\frac{\pi}{2}\right),$$
$$x_2(0) = 1,$$

where the final velocity is not specified.

To solve this problem, we introduce the Hamiltonian

$$H = -u^2 + \lambda_1 \dot{x}_2 + \lambda_2 (u - x_1).$$

Next forming the adjoint and maximum principle equation for this problem, we have

$$\dot{\lambda_1} = \lambda_2,$$
$$\dot{\lambda_2} = -\lambda_1,$$
$$0 = -2u + \lambda_2 \Rightarrow u = \frac{\lambda_2}{2}.$$

We must note here that we also have the transversality condition $\lambda_2\left(\frac{\pi}{2}\right) = 0$.

Applying standard techniques to solve coupled nonlinear differential equations, we have $\lambda_2 = A\cos t + B\sin t$ and applying the transversality condition above implies that $B = 0$, which then enables us to express the control as

$$u = \frac{A}{2}\cos t.$$

Creating a second-order differential equation for x_1 as

$$\ddot{x}_1 = \dot{x}_2 = -x_1 + \frac{A}{2}\cos t,$$

enables us to find general expressions for x_1 and x_2 as

$$x_1 = C_1\cos t + C_2\sin t + \frac{A}{4}t\sin t,$$
$$x_2 = -C_1\sin t + C_2\cos t + \frac{A}{4}(\sin t + t\cos t).$$

Applying the boundary conditions yields $C_1 = 0$, $C_2 = 1$ and $A = -\frac{8}{\pi}$, and so the optimal control and optimal state are

$$u = -\frac{4}{\pi}\cos t,$$
$$x = (1 - 2\pi t)\sin t.$$

7.3 Autonomous (Time-Invariant) Problem

In Chapter 5 we identified three situations that could occur with respect to the functionals we were trying to maximize; these three situations were when either y, $\dot{y}$, or x were missing. If any of these functions of variables were missing, then there were different forms for the Euler equations, specifically when x was missing. For the case when x was missing, we obtained the first integral that introduced a differential equation that needed to be solved. For the optimal control problem we have to consider the case when the functional is independent of t. These types of problems are referred to as **autonomous** or **time invariant**. For the optimal control situation, we start with the following functional problem:

$$\max \int_{t_0}^{t_f} F(x,u)\,dt,$$

subject to

$$\dot{x} = f(x,u),$$
$$x(t_0) = x_0,$$
$$x(t_f) = x_f,$$

where the last boundary condition may or may not be given. As with the other derivations, we form the new functionals

$$\begin{aligned}\max \mathcal{L} &= \int_{t_0}^{t_f} F(x,u) + \lambda\left(f(x,u) - \dot{x}\right) dt, \\ &\equiv \int_{t_0}^{t_f} G\left(x,u,\lambda,\dot{x},\dot{u},\dot{\lambda}\right) dt, \\ &\equiv \int_{t_0}^{t_f} H(x,u,\lambda) - \lambda\dot{x}dt,\end{aligned}$$

that has the associated system of Euler equations

$$\frac{\partial G}{\partial \dot{x}} = -\lambda, \quad \frac{\partial G}{\partial \dot{u}} = 0, \quad \frac{\partial G}{\partial \dot{\lambda}} = 0.$$

Forming the Taylor series expansion of G results in

$$G - \frac{\partial G}{\partial \dot{x}}\dot{x} - \frac{\partial G}{\partial \dot{u}}\dot{u} - \frac{\partial G}{\partial \dot{\lambda}}\dot{\lambda} = H - \lambda\dot{x} + \lambda\dot{x} = constant, \tag{7.32}$$

which implies that the **Hamiltonian is constant along the optimal path**.

Returning to the fishery example, we recall that this problem is autonomous, where we have shown that the Lagrange multiplier is $\lambda = Ke^{-t}$, that the optimal control is $u = -\frac{\lambda}{2} = -\frac{K}{2}e^{-t}$, and that the state is $x = Ce^{t} - \frac{K}{4}e^{-t}$, where the boundary conditions for the initial time and at a known time T will determine the constants C and K. Finally, we know from the state equations that $u = x - \dot{x}$. Forming the Hamiltonian for this problem we have

$$\begin{aligned}H &= -u^2 + \lambda(x-u), \\ &\equiv -\frac{K^2}{4}e^{-2t} + Ke^{-t}\left(Ce^{t} - \frac{K}{4}e^{-t} + \frac{K}{2}e^{-t}\right), \\ &\equiv -\frac{K^2}{4}e^{-2t} + KC + \frac{K^2}{4}e^{-2t} = KC = constant.\end{aligned}$$

A way to think of the meaning of the Hamiltonian being constant is to realize that this condition on the Hamiltonian provides a relationship for the fishery example between $\dot{x}$ and x by recalling that $u = x - \dot{x}$ and $\lambda = -2u$. Therefore, the Hamiltonian for this version of the example is

$$H = -u^2 + \lambda(x-u) \equiv -(x-\dot{x})^2 - 2(x-\dot{x})\dot{x} = \dot{x}^2 - x^2 = KC.$$

Thus, the phase plane associated with this Hamiltonian is a family of hyperbolae with a center at $(0,0)$ and asymptotes at $\dot{x} = \pm x$. An illustration of the phase plane for the current example is shown in Fig. 7.3. The different hyperbolic curves in Fig. 7.3 are determined by the different possible end times T, and as such the curve that gives the optimal control for a specific situation is dependent on the time necessary to complete the transfer from initial state x_0 to the final state x_f.

Exercise 7.5. *Through using calculus of variation techniques, find the extremal of*

$$\int_0^1 \left(12ty + \dot{y}^2\right) dt,$$

satisfying the initial condition of $y(0) = 0$ *and the end condition of* $y(1) = 0$.

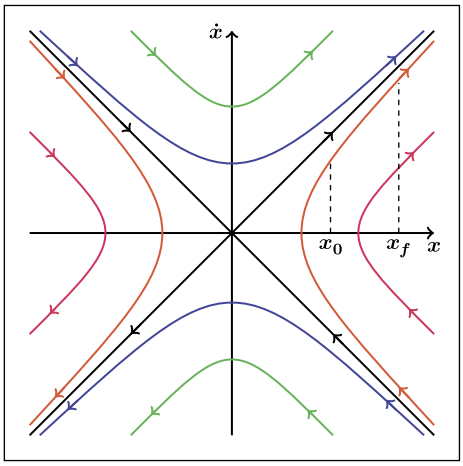


FIGURE 7.3

Illustration of the hyperbolic phase plane.

Exercise 7.6. *Find the extremal of the autonomous functional*

$$\int_{t_0}^{t_1} \left(y^2 - \dot{y}^2\right) dt,$$

satisfying the initial condition of $y\,(t_0) = 0$ *and the end condition of* $y\,(t_1) = \pi$.

Exercise 7.7. *For the second-order control system with state equations*

$$\dot{x} = z, \quad \dot{z} = -z + u,$$

where u *is the control, determine the optimal control that transfers the system from the origin* $x\,(0) = 0$, $z\,(0) = 0$ *at* $t_0 = 0$ *to the position*

$$x\,(1) = \frac{1}{2}\left(e + e^{-1}\right) - 1, \quad z\,(1) = \frac{1}{2}\left(e - e^{-1}\right),$$

at time $t_f = 1$ *and minimizes the cost functional*

$$J = \int_0^1 u^2 dt.$$

7.4 Extension to General Boundary Conditions

We now consider the situation where the control problem does not have a specific set of values for the end time boundary condition, $x\left(t_f\right)$, but a set of functions instead. Therefore, the control problem for

this situation is defined as

$$\begin{aligned} &\max \int_{t_0}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t)\, dt, \\ &\text{subject to} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \quad \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad \boldsymbol{g}\left(\boldsymbol{x}\left(t_f\right), t_f\right) = \boldsymbol{0}, \end{aligned} \tag{7.33}$$

where $\boldsymbol{g} \in \mathbb{R}^p$ for all $\boldsymbol{x}$, t. Thus the problem has p boundary conditions $g_i(\boldsymbol{x}, t) = 0$ at t_f for $i = 1, 2, \ldots, p$.

There are a few assumptions that need to be made before a solution method is used to solve (7.33). The first assumption is that $p \geq n$, while the second assumption is that the set of derivative of the boundary conditions $\left\{\frac{\partial g_j}{\partial \boldsymbol{x}}\right\}$, $j = 1, 2, \ldots, p$, where

$$\frac{\partial g_j}{\partial \boldsymbol{x}} = \begin{bmatrix} \frac{\partial g_1}{\partial \boldsymbol{x}} \\ \frac{\partial g_2}{\partial \boldsymbol{x}} \\ \vdots \\ \frac{\partial g_p}{\partial \boldsymbol{x}} \end{bmatrix},$$

form a set of **independent** vectors, $\forall \boldsymbol{x} \in \mathbb{R}^n$ and $t \in \mathbb{R}$.

The boundary conditions, $g_j(\boldsymbol{x}, t) = g_j(x_1, x_2, \ldots, x_n, t)$, define a **hyper surface** for each $j = 1, 2, \ldots, p$ in $\mathbb{R}^n$, with gradient vectors, $\frac{\partial g_j}{\partial \boldsymbol{x}}$, that are **orthogonal** to the surface at $(\boldsymbol{x}, t)$.

A final assumption that is made is that the intersection of surfaces is a **smooth** manifold on which $\boldsymbol{x}$ must lie at $t = t_f$ and the gradient, $\left\{\frac{\partial g_j}{\partial \boldsymbol{x}}\right\}$, defines a surface that is **orthogonal** to the manifold.

An example of this type of control problem is hitting a specific orbit. If we consider the diagram in Fig. 7.4, we see that we have x_1 and x_2 and we require the states $x_1\left(t_f\right)$ and $x_2\left(t_f\right)$ to lie on a circle. Therefore, the boundary condition at $t = t_f$ is $g(x_1, x_2) = x_1^2 + x_2^2 - r^2 = 0$.

If the boundary conditions do not depend explicitly on t, then the end conditions are **autonomous**, as such the problem involves a moving target.

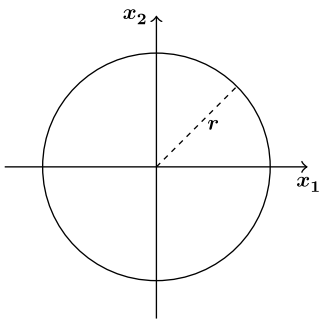


FIGURE 7.4

Illustration of constraint of a circular orbit optimal control problem.

In Chapter 5, we did not consider the case where the boundary condition at the end time or space was a function. As such, we shall briefly summarize the derivation such that we can apply the calculus of variation theory to the control problem stated in (7.33).

7.4.1 Extension of Calculus of Variation Theory

The starting point is to consider the general functional

$$\max \int_a^b G\left(\boldsymbol{y}, \dot{\boldsymbol{y}}, t\right) dt,$$

subject to $\boldsymbol{y}(a) = \boldsymbol{y}_a$ and $\boldsymbol{g}\left(\boldsymbol{y}(b), b\right) = 0$.

Considering the scalar case for the boundary condition, we require variations that lead to necessary conditions for a weak relative optimal. As with the other form of constrained functional problems, we introduce a Lagrange multiplier μ so that the functional now becomes

$$\max \int_a^b G\left(\boldsymbol{y}, \dot{\boldsymbol{y}}, t\right) dt + \mu g\left(\boldsymbol{y}(b), b\right). \tag{7.34}$$

As with other forms of calculus of variation problems that have been considered, we introduce a variation $\eta(t)$, such that

$$\boldsymbol{y}^{\varepsilon}(t) = \boldsymbol{y}(t) + \varepsilon \boldsymbol{\eta}(t), \quad 0 < \varepsilon < 1,$$

where $\boldsymbol{y}(t)$ is the optimal solution and $\boldsymbol{y}(t), \boldsymbol{\eta}(t) \in \mathbb{C}_2[a, b]$. Introducing the variations into (7.34) results in a functional in terms of ε as

$$\psi(\varepsilon) = \int_a^b G\left(\boldsymbol{y}^{\varepsilon}, \dot{\boldsymbol{y}}^{\varepsilon}, t\right) dt + \mu g\left(\boldsymbol{y}^{\varepsilon}(b), b\right).$$

We are seeking $\left.\frac{d\psi}{d\varepsilon}\right|_{\varepsilon=0} = 0$, which can easily be shown to be

$$\begin{aligned}\left.\frac{d\psi}{d\varepsilon}\right|_{\varepsilon=0} = 0 &\equiv \int_a^b \sum_{i=1} n \frac{\partial G}{\partial y_i}\eta_i + \frac{\partial G}{\partial \dot{y}_i}\dot{\eta}_i dt + \mu \sum_{i=1}^{n} \left.\frac{\partial g}{\partial y_i}\right|_{t=b} \eta_i(b), \\ &= \sum_{i=1}^{n} \int_a^b \left(\frac{\partial G}{\partial y_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{y}_i}\right)\right)\eta_i dt + \sum_{i=1}^{n} \left.\frac{\partial G}{\partial \dot{y}_i}\eta_i\right|_{t=b} + \mu \sum_{i=1}^{n} \left.\frac{\partial g}{\partial y_i}\eta_i\right|_{t=b}.\end{aligned} \tag{7.35}$$

Applying Lagrange's lemma to (7.35) results in the system of Euler equations for each y_i as

$$\frac{\partial G}{\partial y_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{y}_i}\right) = 0, \quad i = 1, 2, \ldots, n. \tag{7.36}$$

As the boundary condition $t = b$ is a function, it is not possible to use the technique used throughout Chapter 5 where $\eta_i(b) = 0$ $i = 1, 2, \ldots, n$. Thus we require transversality conditions that follow the techniques shown in Chapter 5, that results in

$$\left(\frac{\partial G}{\partial \dot{y}_i} + \mu \frac{\partial g}{\partial y_i}\right)\Bigg|_{t=b} = 0, \quad i = 1, 2, \ldots, n. \tag{7.37}$$

Therefore, to find the weak relative optimal for (7.33), we have to solve (7.36) and (7.37).

Next, we consider the situation of multiple boundary conditions, $g_j(\boldsymbol{y}(b), b)$, $j = 1, 2, \ldots, p$ or $\boldsymbol{g}(\boldsymbol{y}(b), b)$. This results in the functional problem

$$\max \int_a^b G(\boldsymbol{y}, \dot{\boldsymbol{y}}, t)\, dt + \sum_{j=1}^{p} \mu_j g_j(\boldsymbol{y}(b), b), \tag{7.38}$$

where there are now p Lagrange multipliers, μ_j.

Applying the same calculus of variation techniques to (7.38) results in the same system of Euler equations as in (7.36), but now the system of transversality conditions become

$$\left(\frac{\partial G}{\partial \dot{y}_i} + \sum_{j=1}^{p} \mu_j \frac{\partial g_j}{\partial y_i}\right)\Bigg|_{t=b} = 0, \qquad \begin{matrix} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, p \end{matrix}. \tag{7.39}$$

Given the equations needed to solve a functional problem with a function as the end boundary condition, we now return to the optimal control problem to derive expressions to find the equivalent equations for these types of control problems.

7.4.2 Optimal Control Problems With General Boundary Conditions

Given the constraints and the general boundary conditions presented in the last subsection, the equivalent optimal control problem becomes

$$\max \int_{t_0}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T (\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) - \dot{\boldsymbol{x}})\, dt + \boldsymbol{\mu}^T \boldsymbol{g}(\boldsymbol{x}(t_f), t_f). \tag{7.40}$$

As with the previous optimal control functional, we introduce the function $G \equiv F + \boldsymbol{\lambda}^T (\boldsymbol{f} - \dot{\boldsymbol{x}})$, and the Hamiltonian $H = F + \boldsymbol{\lambda}^T \boldsymbol{f}$. We already have expressions for the system of Euler equations in terms of the Hamiltonian, so we focus here on the new transversality conditions that arise with general boundary conditions. From the calculus of variation derivation in the previous subsection, we have the set of conditions

$$\left(\frac{\partial G}{\partial \dot{y}_i} + \sum_{j=1}^{p} \mu_j \frac{\partial g_j}{\partial y_i}\right)\Bigg|_{t=b} = 0, \quad i = 1, 2, \ldots, n,$$

where for the control problem we have $\dot{x}_i$ and x_i and it has been shown that $\frac{\partial G}{\partial \dot{x}_i} = -\lambda_i$. Therefore the transversality condition for (7.40) becomes

$$\lambda_i(t_f) = \sum_{j=1}^{p} \mu_j \frac{\partial g_j}{\partial x_i}\Bigg|_{t=t_f}. \tag{7.41}$$

Thus the transversality condition in (7.41) implies that the first Lagrange multipliers, λ_i, are linear combinations of columns that are orthogonal to the surface defined by $\boldsymbol{g}$ at $t = t_f$.

To help illustrate this theory, we return to the mass on a spring example:

Example 7.8. *Consider the optimal control problem* $\min \int_0^{\frac{\pi}{2}} u^2 dt \equiv \min \int_0^{\frac{\pi}{2}} -u^2 dt$ *subject to* $\dot{x}_1 = x_2$, $\dot{x_2} = -x_1 + u$, *with initial conditions* $x_1(0) = 0$ *and* $x_2(0) = 1$ *and the function at* $t = \frac{\pi}{2}$ *is* $g \equiv x_1\left(\frac{\pi}{2}\right) - \alpha x_2\left(\frac{\pi}{2}\right) = 0$.

We have already solved for the system of Euler equations and have the expressions

$$\begin{aligned} \dot{\lambda_1} &= \lambda_2, \\ \dot{\lambda_2} &= -\lambda_1, \\ u &= \frac{\lambda_2}{2}. \end{aligned}$$

Evaluating the transversality condition for this problem results in

$$\boldsymbol{\lambda}(t_f) = \mu \left.\frac{\partial g}{\partial \boldsymbol{x}}\right|_{t=t_f} \Rightarrow \begin{pmatrix} \lambda_1(t_f) \\ \lambda_2(t_f) \end{pmatrix} = \mu \begin{pmatrix} 1 \\ -\alpha \end{pmatrix}. \tag{7.42}$$

We eliminate μ by noticing that $\lambda_1\left(\frac{\pi}{2}\right) = \mu$, so that $\lambda_2\left(\frac{\pi}{2}\right) - \mu\alpha = -\alpha\lambda_1\left(\frac{\pi}{2}\right)$, which results in the boundary condition for the λ_is as $\alpha\lambda_1\left(\frac{\pi}{2}\right) + \lambda_2\left(\frac{\pi}{2}\right) = 0$. Therefore, there are four differential equations with four boundary conditions to solve, which implies that there is a **unique solution**.

The first differential equation $\ddot{\lambda}_2 = -\lambda_2 \Rightarrow \lambda_2 = A\cos t + B\sin t$ and $\lambda_1 = -\dot{\lambda}_2 = A\sin t - B\cos t$. Next, applying the boundary condition above, we have $\alpha\lambda_1\left(\frac{\pi}{2}\right) + \lambda_2\left(\frac{\pi}{2}\right) = 0 \Rightarrow \alpha A + B = 0$. We now evaluate the condition linking the control, u, to the Lagrange multiplier, λ_2 as $u = \frac{\lambda_2}{2} = \frac{A}{2}(\cos t - \alpha \sin t)$.

Returning to the state equations, we have $\ddot{x}_1 + x_1 = \frac{A}{2}(\cos t - \alpha \sin t)$, $x_1(0) = 0$ and $x_2(0) = 1$. Therefore, the general expression for the state variable is

$$x_1 = \left(1 - \frac{\alpha A}{4} + \frac{A}{4}t\right)\sin t + \frac{\alpha A}{4} t\cos t.$$

Finally using the last boundary condition, $x_1\left(\frac{\pi}{2}\right) - \alpha x_2\left(\frac{\pi}{2}\right) = 0$, results in an expression for A as

$$A = \frac{8}{4\alpha - (1+\alpha^2)\pi}.$$

7.5 Free End Time Optimal Control Problems

We now consider the situation where t_f is not specified; therefore, as well as finding the optimal control, we also have to find the optimal final time. As in the last subsection, we shall apply the calculus of variation approach to these types of problems first before extending the theory to the optimal control situations.

7.5.1 Extension of the Calculus of Variation Theory

We are therefore seeking extremals of the functional

$$\max \int_a^b G(\mathbf{y}, \dot{\mathbf{y}}, t)\, dt + \mu g(\mathbf{y}(b), b), \quad \mathbf{y}(a) = \mathbf{y}_a, \tag{7.43}$$

where b is unknown.

In Fig. 7.5, we have an illustration of the situations that could occur for these types of problems. In this figure we have two solutions that satisfy the initial condition, but both solutions also satisfy the end boundary condition value though at times b and at $b + \delta b$. Therefore, in order to be able to solve this type of problem, we shall have to vary both the functional and the interval.

We start by assuming that $\mathbf{y}(t)$ is optimal in the interval $[a, b]$, and $\mathbf{y} \in \mathbb{C}_2[a, b]$. Next we consider a general variation that is dependent on the parameter $0 < \varepsilon < 1$, denoted by $\tilde{\mathbf{y}}(t, \varepsilon) \in \mathbb{C}_2[a, b(\varepsilon)]$, with the property $\tilde{\mathbf{y}}(t, 0) = \mathbf{y}(t)$. The next set of conditions on the variation are

$$\frac{\partial \tilde{\mathbf{y}}}{\partial t}(t, 0) = \dot{\mathbf{y}}(t), \quad \varepsilon = 0, \tag{7.44}$$

$$\tilde{\mathbf{y}}(a, \varepsilon) = \mathbf{y}_a, \quad \forall \varepsilon, \tag{7.45}$$

$$\tilde{\mathbf{y}}(b(\varepsilon), \varepsilon), \quad \text{satisfies} \quad g(\tilde{\mathbf{y}}(b(\varepsilon), \varepsilon)) = 0. \tag{7.46}$$

In the following derivation we shall use $b(\varepsilon)|_\varepsilon$ to denote b at $\varepsilon = 0$. We shall also use the fact that $b(\varepsilon) = b + \varepsilon \frac{db}{d\varepsilon} + O(\varepsilon^2)$.

The next step is to define the functional problem as

$$\psi(\varepsilon) = \int_a^{b(\varepsilon)} G\left(\tilde{\mathbf{y}}(t, \varepsilon), \frac{\partial \tilde{\mathbf{y}}(t, \varepsilon)}{\partial t}, t\right) dt + \mu g(\tilde{\mathbf{y}}(b(\varepsilon), \varepsilon), b(\varepsilon)), \tag{7.47}$$

where we require $\left.\frac{\partial \psi}{\partial \varepsilon}\right|_{\varepsilon=0} = 0$ for the solution to be optimal. In order to treat the derivatives of the variable interval, we require **Leibnitz's Rule**, which states that

$$\frac{d}{d\varepsilon} \int_a^{b(\varepsilon)} f(t)\, dt = f(b) \frac{db}{d\varepsilon}. \tag{7.48}$$

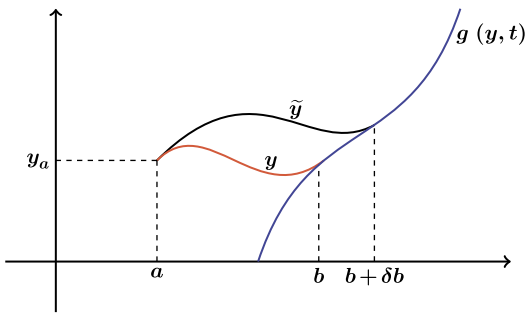


FIGURE 7.5

Illustration of the situation where the end condition is a function.

Proof. Define the functionals $I(0)$ and $I(\varepsilon)$ to be

$$I(0) \equiv \int_a^b f(t)\,dt, \qquad I(\varepsilon) \equiv \int_a^{b(\varepsilon)} f(t)\,dt,$$

respectively. Now, considering the definition of the gradient operator as the limit as $\varepsilon \to 0$, we have

$$\begin{aligned}
\left.\frac{dI}{d\varepsilon}\right|_{\varepsilon=0} &\equiv \lim_{\varepsilon\to 0}\frac{I(\varepsilon)-I(0)}{\varepsilon},\\
&= \lim_{\varepsilon\to 0}\frac{1}{\varepsilon}\int_b^{b(\varepsilon)} f(t)\,dt,\\
&= \lim_{\varepsilon\to 0}\frac{b(\varepsilon)-b}{\varepsilon} f\left(b+\theta\left(b(\varepsilon)-b\right)\right), \quad 0<\theta<1,\\
&= f(b)\frac{db}{d\varepsilon},
\end{aligned}$$

where we have used the **mean value theorem** to move from line 2 to line 3 of the proof above, and have taken the limit as ε tends to zero to obtain the final expression, which completes the proof.

Before returning to (7.47) we have the following three definitions to consider:

$$\eta_i(t) \equiv \left.\frac{\partial \tilde{y}_i(t,\varepsilon)}{\partial t}\right|_{\varepsilon=0}, \; \dot{\eta}_i(t) \equiv \left.\frac{\partial}{\partial t}\left(\frac{\partial \tilde{y}_i(t,\varepsilon)}{\partial t}\right)\right|_{\varepsilon=0} \equiv \left.\frac{\partial}{\partial \varepsilon}\left(\frac{\partial \tilde{y}_i(t,\varepsilon)}{\partial t}\right)\right|_{\varepsilon=0}. \tag{7.49}$$

Taking the first variation of (7.47) with respect to ε yields

$$\psi'\big|_\varepsilon = \int_a^b \sum_{i=1}^n \left(\frac{\partial G}{\partial y_i}\eta_i + \frac{\partial G}{\partial \dot{y}_i}\dot{\eta}_i\right)dt + G|_b \frac{db}{d\varepsilon} + \mu \sum_{i=1}^n \left(\frac{\partial g}{\partial y_i}\frac{d\tilde{y}_i}{d\varepsilon} + \frac{\partial g}{\partial t}\frac{db}{d\varepsilon}\right)\Bigg|_{\substack{t=b\\ \varepsilon=0}},$$

where

$$\begin{aligned}
\left.\frac{d\tilde{y}_i}{d\varepsilon}\right|_{\substack{t=b\\ \varepsilon=0}} &= \left.\frac{\partial \tilde{y}_i}{\partial t}\frac{db}{d\varepsilon} + \frac{\partial \tilde{y}_i}{\partial \varepsilon}\right|_{\substack{t=b\\ \varepsilon=0}},\\
&= \dot{y}_i \frac{db}{d\varepsilon} + \eta_i|_{t=b}.
\end{aligned}$$

Using partial integration, and the expression above, results in

$$\begin{aligned}
\psi'\big|_{\varepsilon=0} = \int_a^b \sum_{i=1}^n \left(\frac{\partial G}{\partial y_i} - \frac{d}{dt}\left(\frac{dG}{d\dot{y}_i}\right)\right)\eta_i dt + \sum_{i=1}^n \left.\frac{\partial G}{\partial \dot{y}_i}\eta_i\right|_{t=b} + G|_{t=b}\frac{db}{d\varepsilon}\\
+ \mu \sum_{i=1}^n \left.\frac{\partial g}{\partial y_i}\eta_i\right|_{t=b} + \mu \sum_{i=1}^n \left.\frac{\partial g}{\partial y_i}\dot{y}_i\right|_{t=b}\frac{db}{d\varepsilon} + \mu \left.\frac{\partial g}{\partial t}\right|_{t=b}\frac{db}{d\varepsilon}.
\end{aligned}$$

Applying Lagrange's lemma to the integral above results in the series of Euler equations; collecting the terms that are multiples of $\eta_i|_{t=b}$ results in the first transversality conditions; and collecting the terms

$\frac{db}{d\varepsilon}$, such that the terms are all equal to zero, results in a second transversality condition:

$$\sum_{i=1}^{n} \frac{\partial G}{\partial y_i} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{y}_i}\right) = 0, \tag{7.50}$$

$$\sum_{i=1}^{n} \frac{\partial G}{\partial \dot{y}_i}\bigg|_{t=b} + \mu \sum_{i=1}^{n} \frac{\partial g}{\partial y}\bigg|_{t=b} = 0, \tag{7.51}$$

$$G|_{t=b} - \sum_{i=1}^{n} \frac{\partial G}{\partial \dot{y}_i}\dot{y}_i\bigg|_{t=b} + \mu \left.\frac{\partial g}{\partial t}\right|_{t=b} = 0. \tag{7.52}$$

The theory above is easily extendable to multiple conditions $\boldsymbol{g}\left(\boldsymbol{y}\left(b\right), b\right)$ where (7.52) would become

$$G|_{t=b} - \sum_{i=1}^{n} \frac{\partial G}{\partial \dot{y}_i}\dot{y}_i\bigg|_{t=b} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t=b} = 0. \tag{7.53}$$

7.5.2 Applying the Theory to Control Problems

Maximize the cost functional

$$\max \mathcal{L} = \int_{t_0}^{t_f} F\left(\boldsymbol{x}, \boldsymbol{\mu}, t\right) + \boldsymbol{\lambda}^T \left(\boldsymbol{f}\left(\boldsymbol{x}, \boldsymbol{u}, t\right) - \dot{\boldsymbol{x}}\right) dt, \tag{7.54}$$

subject to

$$\dot{\boldsymbol{x}} = \boldsymbol{f}\left(\boldsymbol{x}, \boldsymbol{u}, t\right), \quad \boldsymbol{x}\left(t_0\right) = \boldsymbol{x}_0, \quad \boldsymbol{g}\left(\boldsymbol{x}\left(t_f\right), t_f\right), \tag{7.55}$$

where t_f is not defined.

As before, we define the function $G \equiv F + \boldsymbol{\lambda}^T \boldsymbol{f} - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}} = H - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}}$ and apply the calculus of variation techniques to (7.54), which results in the system of differential equations derived earlier along with the transversality condition. We now focus on the second transversality condition, where we have

$$G|_{t_f} + \sum_{i=1}^{n} \lambda_i \dot{x}_i\bigg|_{t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t_f} = 0,$$

$$H|_{t_f} - \sum_{i=1}^{n} \lambda_i \dot{x}_i\bigg|_{t_f} + \sum_{i=1}^{n} \lambda_i \dot{x}_i|_{t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t_f} = 0,$$

$$H|_{t_f} = -\sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t_f}. \tag{7.56}$$

There are some special cases that we have to consider: (1) If $\boldsymbol{g}\left(\boldsymbol{x}\left(t_f\right)\right)$, or $\boldsymbol{g} = \boldsymbol{g}\left(\boldsymbol{x}\right)$ at $t = t_f$, that is to say, if the end conditions are homogeneous/autonomous so that they do not depend on t explicitly, then we have that

$$H|_{t_f} = 0. \tag{7.57}$$

(2) If it is the case that the control problem is also autonomous, then the system $\boldsymbol{f}$ and F do not depend explicitly on t, and as such H is equal to a constant on the optimal trajectory, which has been shown in previous arguments using the integral form of the Euler equation, and given the condition in (7.57), then along the optimal trajectory we have

$$H = 0, \quad \forall t. \tag{7.58}$$

To help illustrate all this theory that has just been derived, we return to the fishery example. We know that this problem is autonomous, but now we have the end condition, $g = x(T) - x_T = 0$, where T is not specified. We have already shown that the state x is defined by $x = Ce^t - \frac{K}{4}e^{-t}$. We now have to solve the second transversality condition $H|_T = 0$. However, as the system is autonomous, we have to solve the condition $H = 0\ \forall t$ on the optimal trajectory. It has already been proven that $H = KC$, which we require to be equal to zero, which implies that K or C is equal to zero.

Considering the two possible outcomes, we have

$$\begin{aligned} K = 0 \quad &\Rightarrow \quad x = Ce^t \quad &&\Rightarrow \quad \text{increasing}, \\ C = 0 \quad &\Rightarrow \quad x = -\tfrac{K}{4}e^{-t} \quad &&\Rightarrow \quad \text{decreasing}. \end{aligned}$$

Hence, we have two possible situations, that are:

(a) $x_t > x_0 > 0$ or $(x_T < x_0 < 0)$ then set $K = 0$. To satisfy the initial condition $x(0) = x_0 \Rightarrow C = x_0$. Therefore, we have that $x = x_0 e^t$ but also that $u = 0$. The final condition $x(T) = x_0 e^T = x_T \Rightarrow T = \ln\left(\frac{x_T}{x_0}\right) > 0$.

(b) $0 < x_T < x_0$ or $(x_0 < x_T < 0)$ then set $C = 0$ To satisfy the initial condition $x(0) = x_0 \Rightarrow K = -4x_0$. Therefore, we have that $x = x_0 e^{-t}$ but now the control u is equivalent to $u = \frac{-K}{2}e^{-t} \equiv 2x_0 e^{-t} = 2x$. The final condition $x(T) = x_0 e^{-T} = x_T \Rightarrow T = \ln\left(\frac{x_0}{x_T}\right) > 0$.

Thus, this is the synthesis of control, that is to say, given the goal, x_T, (compared to x_0), then given any state, we know the associated optimal control to follow.

Now we consider the example where $x, \lambda \in \mathbb{D}_1$ and $u \in \mathbb{D}_0$, with the functional

$$\min \int_{t_0}^{t_f} x^2 (1+u)^2 \, dt, \quad \dot{x} = u, \quad x(0) = 0, \quad x(2) = 1. \tag{7.59}$$

Applying the theory presented implies that $x^2 - c = (1+k)^2$, but there is no value for C and K that satisfies the initial and end conditions. There is, however, a solution that is piecewise smooth. Now we change the problem to be

$$\int_{t_0}^{t_f} x^2 (1-u)^2 \, dt = 0,$$

if either $x = 0$ or $u = 1 = \dot{x}$, then the optimal solution is

$$\begin{cases} x = 0 & 0 \le t \le 1 \\ u = 0 \end{cases}$$

$$\begin{cases} x = t - 1 & 1 \le t \le 2 \\ u = 1 \end{cases}$$

Exercise 7.9. *Find a solution to the problem* $\min \int_0^T u^2 dt$ *subject to* $\dot{x} = y - x$, $\dot{y} = u - 2y$ *with the initial conditions* $x(0) = 0$, $y(0) = 0$, *and the end condition function* $x(T) + y(T) = 2$. *In the case that* T *is given, show that* $u = \frac{2e^t}{\sinh T}$. *In the case where* T *is not given, under what conditions must* T *satisfy?*

7.6 Piecewise Smooth Calculus of Variation Problems

As with other theories beyond that presented in Chapter 5, we shall state and prove the new calculus of variation theory and then apply it to the control problem.

7.6.1 Extension of Calculus of Variation Techniques

We start with the functional

$$\max \int_a^b G(y, \dot{y}, t)\, dt, \quad \text{subject to} \quad y(a) = y_a, \quad [y(b) = y_b],$$

over all functions $y \in \mathbb{D}_1[a, b]$, where, as we introduced at the beginning of this chapter, $\mathbb{D}_1$ is the space of piecewise smooth functions. We also have that G is differentiable with respect to y, $\dot{y}$, and t. We now seek necessary conditions for a weak relative maximum or minimum (stationary point).

As with the other derivations, we look at variations of the form

$$y^\varepsilon = y + \varepsilon\eta,$$

where y is the optimal solution and $\eta \in \mathbb{D}_1[a, b]$ with $\eta(a) = 0$ and $\eta(b) = 0$ if the end boundary condition is given.

To be able to solve this type of functional problem, we require the following theorem.

Lemma 7.10. ***Du Bois-Reymond lemma:*** *If* $M(t)$ *is piecewise continuous, which implies that the function is bounded on* $[a, b]$ *and continuous at all but a finite number of points, where it may have jump discontinuity and if*

$$\int_a^b M(t)\dot{\eta}(t) = 0, \quad \forall \eta \in \mathbb{D}_1[a, b],$$

such that $\eta(a) = 0$ *and* $\eta(b) = 0$, *then there exists a constant* C *such that* $M(t) = C$ *at every point* $t \in [a, b]$ *where* $M(t)$ *is continuous.*

Returning to the problem at hand, we now introduce a variation to the function y in the standard form $y^\varepsilon = y + \varepsilon\eta$ but now $\eta \in \mathbb{D}_1[a, b]$, but still with $\eta(a) = 0$ and $\eta(b) = 0$ if $y(b)$ is given. We again define the new functional $\psi(\varepsilon)$ as

$$\psi(\varepsilon) = \int_a^b G(y + \varepsilon\eta, \dot{y} + \varepsilon\dot{\eta}, t)\, dt,$$

and still require

$$\left.\frac{d\psi}{d\varepsilon}\right|_{\varepsilon=0}.$$

To be able to solve for the optimal solution, we break up the interval $[a, b]$, into subintervals over which there is continuity and differentiability of G, and then add these solutions together. Therefore,

$$\psi'|_{\varepsilon=0} \equiv \left.\frac{d\psi}{d\varepsilon}\right|_{\varepsilon=0} \equiv \int_a^b \left(\frac{\partial G}{\partial y}\eta + \frac{\partial G}{\partial \dot{y}}\dot{\eta}\right) dt = 0, \tag{7.60}$$

where y and $\dot{y}$ are evaluated at the optimal.

Unlike in the previous calculus of variation derivations, we do not apply partial integration to $\frac{\partial G}{\partial \dot{y}}\dot{\eta}$ in (7.60), instead we apply partial integration to the other term in (7.60) as follows: Introduce a new function, $\chi\,(t)$, defined as

$$\chi\,(t) = \int_a^t \frac{\partial G}{\partial y} ds, \tag{7.61}$$

then $\chi\,(a) = 0$, and

$$\dot{\chi} = \frac{\partial G}{\partial y}, \tag{7.62}$$

except at corners where $\frac{\partial G}{\partial y}$ is not continuous. Substituting (7.62) into (7.60) results in

$$\begin{aligned} \psi'|_{\varepsilon=0} &= \int_a^b \left(\frac{\partial G}{\partial y}\eta + \frac{\partial G}{\partial \dot{y}}\dot{\eta}\right) dt, \\ &= \int_a^b \left(\dot{\chi}\eta + \frac{\partial G}{\partial \dot{y}}\dot{\eta}\right) dt, \end{aligned} \tag{7.63}$$

and now apply integration by parts to the first term in the integral in (7.63), which results in

$$\psi'|_{\varepsilon=0} = \chi\eta|_a^b + \int_a^b \left(-\chi + \frac{\partial G}{\partial \dot{y}}\right)\dot{\eta} dt = 0, \quad \forall \eta \in \mathbb{D}_1\,[a, b]\,,\ \eta\,(a) = \eta\,(b) = 0. \tag{7.64}$$

Applying Lagrange's lemma to (7.64) implies that $-\chi + \frac{\partial G}{\partial \dot{y}} = 0$. Now applying the Du-Bois-Reymond lemma yields

$$\begin{aligned} \frac{\partial G}{\partial \dot{y}} &= \chi + c, \\ &= \int_a^t \frac{\partial G}{\partial y} ds + c. \end{aligned} \tag{7.65}$$

An important feature to note here about (7.65) is that it is the **integrated** form of the Euler equation. Therefore, if we differentiate (7.65) over the intervals where the functions are continuous, then we obtain

$$\frac{\partial G}{\partial y} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{y}}\right) = 0,$$

which implies that the Euler equation holds except at a finite number of points where the optimal solution has *corners*.

Consider the case now where $y(b)$ is not given; this implies that $\eta(b)$ can vary. Using the fact that $\frac{\partial G}{\partial \dot{y}} - \chi = c$, we can rewrite (7.64) as

$$\begin{aligned}
\psi'|_{\varepsilon=0} &= \chi(b)\eta(b) + \int_a^b c\dot{\eta}dt = 0, \\
&= \chi(b)\eta(b) + c(\eta(b) - \eta(a)) = 0, \\
&= (\chi(b) + c)\eta(b) = 0, \ \forall \eta(b), \\
&\Rightarrow \chi(b) + c = 0, \\
&\Rightarrow \left.\frac{\partial G}{\partial \dot{y}}\right|_b = 0,
\end{aligned} \tag{7.66}$$

where (7.66) is the transversality condition and comes from using a rearrangement of the first line in (7.65).

We now consider a new condition, referred to as a corner condition. We start by noting that $\dot{y}$ is discontinuous at corners, but we have that

$$\chi(t) = \int_a^b \frac{\partial G}{\partial y} dt,$$

is continuous, i.e., $\chi \in \mathbb{D}_1[a, b]$, which implies that

$$\frac{\partial G}{\partial \dot{y}} = \chi + c, \tag{7.67}$$

is continuous at corners. The condition defined by (7.67) is referred to as the **first Weierstrass-Erdmann corner condition**. The interpretation of this condition is that even at points where $\dot{y}$ is discontinuous, we have

$$\dot{y}(ts^-) = p_1, \quad \dot{y}(ts^+) = p_2, \quad p_1 \neq p_2, \tag{7.68}$$

then

$$\frac{\partial G}{\partial \dot{y}}(y, p_1, ts) = \frac{\partial G}{\partial \dot{y}}(y, p_2, ts),$$

and therefore $\frac{\partial G}{\partial \dot{y}}$ is continuous.

There is also a **second Weierstrass corner condition**, which is given by

$$G - \frac{\partial G}{\partial \dot{y}}\dot{y}, \tag{7.69}$$

which is continuous at corners.

The results that have just been shown for the scalar case can easily be extended to the multivariable case as

$$\begin{aligned}
\frac{\partial G}{\partial \mathbf{y}} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\mathbf{y}}}\right) = 0, \\
\left.\frac{\partial G}{\partial \mathbf{y}}\right|_{t=b} = 0,
\end{aligned}$$

$$\frac{\partial G}{\partial \dot{\boldsymbol{y}}} \quad \text{continuous,}$$
$$G - \dot{\boldsymbol{y}}^T \frac{\partial G}{\partial \dot{\boldsymbol{y}}} \quad \text{continuous.}$$

7.6.2 Application to the Optimal Control Problem

Given the calculus of variation theory and results just proven, we apply this theory to our general multivariable optimal control problem, which is stated as

$$\max \int_{t_0}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t)\, dt, \quad \text{subject to} \quad \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t),$$
$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad \left[\boldsymbol{x}(t_f) = \boldsymbol{x}_f\right].$$

Creating the functional $\mathcal{L}$, as we have done for the other situations, results in

$$\max \mathcal{L} = \int_{t_0}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T (\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) - \dot{\boldsymbol{x}})\, dt,$$
$$\equiv \int_{t_0}^{t_f} H(\boldsymbol{x}, \boldsymbol{u}, t) - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}} dt.$$

We now consider the case where we wish to optimize over $\boldsymbol{x} \in \mathbb{D}_1\left[t_0, t_f\right]$ and $\boldsymbol{u} \in \mathbb{D}_0\left[t_0, t_f\right]$.

To apply the theory that we have just derived, we introduce an augmented vector defined by

$$\boldsymbol{y} \equiv \begin{pmatrix} \boldsymbol{x} \\ \dot{\boldsymbol{U}} \\ \boldsymbol{\lambda} \end{pmatrix}, \tag{7.70}$$

where

$$\boldsymbol{U}(t) = \int_{t_0}^{t_f} \boldsymbol{u} dt, \tag{7.71}$$

that has the properties: $\boldsymbol{U}(t_0) = \boldsymbol{0}$, $\dot{\boldsymbol{U}} = \boldsymbol{u}$ and $\boldsymbol{U} \in \mathbb{D}_1\left(t_0, t_f\right)$. Next we redefine the G function in terms of the components of $\boldsymbol{y}$ as

$$G\left(\boldsymbol{x}, \boldsymbol{U}, \boldsymbol{\lambda}, \dot{\boldsymbol{x}}, \dot{\boldsymbol{U}}, \dot{\boldsymbol{\lambda}}, t\right) \equiv H\left(\boldsymbol{x}, \dot{\boldsymbol{U}}, \boldsymbol{\lambda}, t\right) - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}}. \tag{7.72}$$

The necessary condition are

- **Euler equations:**

$$\frac{\partial G}{\partial \boldsymbol{\lambda}} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\boldsymbol{\lambda}}}\right) = 0 \Rightarrow \boldsymbol{f} = \dot{\boldsymbol{x}}$$
$$\frac{\partial G}{\partial \boldsymbol{\lambda}} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\boldsymbol{\lambda}}}\right) = 0 \Rightarrow \dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \boldsymbol{x}},$$
$$\frac{\partial G}{\partial \boldsymbol{\lambda}} - \frac{d}{dt}\left(\frac{\partial G}{\partial \dot{\boldsymbol{\lambda}}}\right) = 0 \Rightarrow \frac{\partial H}{\partial \boldsymbol{u}} = c,$$

- **Corner condition 1:** The term $\frac{\partial G}{\partial \dot{\boldsymbol{x}}}$ is continuous implies that the Lagrange multiplier, $\boldsymbol{\lambda}$, is continuous.
- **Corner condition 2:** The term $\frac{\partial G}{\partial \dot{\boldsymbol{U}}}$ is continuous implies that the $\frac{\partial H}{\partial \boldsymbol{u}}$ is constant $\forall t \in \left[t_0, t_f\right]$.
- **Transversality condition 1:**

$$\left.\frac{\partial G}{\partial \dot{\boldsymbol{x}}}\right|_{t=t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial \boldsymbol{x}}\right|_{t=t_f} = 0 \Rightarrow \boldsymbol{\lambda}\left(t_f\right) = \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial \boldsymbol{x}}\right|_{t=t_f}.$$

  This is implying that $\boldsymbol{\lambda}\left(t_f\right)$ is in the space spanned by the columns $\left.\frac{\partial g_j}{\partial \boldsymbol{x}}\right|_{t=t_f}$ which define a plane $\perp$ (perpendicular) to the hyperplane where $\boldsymbol{x}\left(t_f\right)$ lies.
- **Transversality condition 2:** If the final time t_f is not specified, then we have

$$G|_{t=t_f} - \sum_{i=1}^{n} \left.\frac{\partial G}{\partial \dot{x}_i}\dot{x}_i\right|_{t=t_f} + \sum_{i=1}^{n} \left.\frac{\partial G}{\partial \dot{U}}\dot{U}\right|_{t=t_f} + \sum_{i=1}^{n} \left.\frac{\partial G}{\partial \dot{\lambda}_i}\dot{\lambda}_i\right|_{t=t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t=t_f} = 0,$$

$$H|_{t=t_f} - \sum_{i=1}^{n} \lambda_i \dot{x}_i\Big|_{t=t_f} + \sum_{i=1}^{n} \lambda_i \dot{x}_i\Big|_{t=t_f} + \sum_{i=1}^{n} \left.\frac{\partial H}{\partial u_i}u_i\right|_{t=t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t=t_f} = 0,$$

  but

$$\left.\frac{\partial G}{\partial \dot{U}}\right|_{t=t_f} + \sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial \dot{U}}\right|_{t=t_f} = 0 \Rightarrow \left.\frac{\partial H}{\partial u}\right|_{t=t_f} = 0,$$

  together with $\frac{\partial H}{\partial \boldsymbol{u}} = c \Rightarrow \frac{\partial H}{\partial \boldsymbol{u}} = \boldsymbol{0}$, $\forall t$, implies that

$$H|_{t=t_f} = -\sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t=t_f},$$

  if t_f is not given.
  For the autonomous case where H and $\boldsymbol{g}$ do not depend explicitly on t, we have

$$H|_{t=t_f} = 0,$$

  if t_f is not specified, $H = a$. Thus $H \equiv 0$, if t_f is not given.

For J to have a strong relative stationary point, maximum, at $\boldsymbol{x}$, $\boldsymbol{u}$ over all admissible functions, $\boldsymbol{u} \in \mathbb{D}_0\left[t_0, t_f\right]$, $\boldsymbol{x} \in \mathbb{D}_0\left[t_0, t_f\right]$ then it is necessary that a continuous n-dimensional function $\boldsymbol{\lambda}(t) \in \mathbb{D}_1\left[t_0, t_f\right]$ exists, not identically zero, and a constant p-dimensional vector $\boldsymbol{\mu}$ such that if

$$H = F(\boldsymbol{x}, \boldsymbol{x}, t) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t),$$

then $\left.\begin{array}{l}\dot{\boldsymbol{x}}=\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},t)\\ \dot{\boldsymbol{\lambda}}=-\frac{\partial H}{\partial \boldsymbol{x}}\end{array}\right\}$ except at the corners of $\boldsymbol{x}$, and where $\boldsymbol{x}$, $\boldsymbol{\lambda}$ are continuous on $\left[t_0,t_f\right]$ with $\frac{\partial H}{\partial \boldsymbol{y}}=\boldsymbol{0}$, and $\frac{\partial H}{\partial \boldsymbol{u}}$ is continuous, $\boldsymbol{x}(t_0)=\boldsymbol{x}_0$, $\boldsymbol{g}\left(\boldsymbol{x}\left(t_f\right),t_f\right)=0$, $\boldsymbol{\lambda}\left(t_f\right)=\sum_{j=1}^{p}\mu_j\left.\frac{\partial g_j}{\partial \boldsymbol{x}}\right|_{t=t_f}$, $H|_{t=t_f}=-\sum_{j=1}^{p}\mu_j\left.\frac{\partial g_j}{\partial t}\right|_{t=t_f}$ if t_f is not specified.

The condition given above are those for a strong relative stationary value, where a stationary value is the cost function evaluated at the optimal control.

This now enables us to state the following corollary:

Corollary 7.11. *If a system is autonomous, that is $F=F(\boldsymbol{x},\boldsymbol{u})$, $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})$, then H equals a constant $\forall t$. If the boundary/end conditions are autonomous, that is $\boldsymbol{g}=\boldsymbol{g}\left(t_f\right)$, then $H|_{t=t_f}=0$ if t_f is not given. If both the system and the boundary/end conditions are autonomous, then $H\equiv 0\ \forall t$ if t_f is not specified.*

To help illustrate all of the theory that has just been presented, we consider the following example:

Example 7.12. ***Time minimization:*** *Minimize the functional*

$$\min T=\int_0^T 1dt, \tag{7.73}$$

subject to

$$\dot{x}=y, \qquad x(1)=0,\ y(0)=0, \tag{7.74}$$

$$\dot{y}=ue^{\frac{1}{2}\left(1-u^2\right)}, \qquad x(T)=1,\ y(T)=0. \tag{7.75}$$

Forming the Hamiltonian $H=-1+\lambda_1 y+\lambda_2 ue^{\frac{1}{2}(1-u^2)}$ leads to the adjoint equations

$$\dot{\lambda}_1=-\frac{\partial H}{\partial x}=0\Rightarrow\lambda_1=c_1,$$

$$\dot{\lambda}_2=-\frac{\partial H}{\partial y}=-\lambda_1=-c_1\Rightarrow\lambda_2=c_2-c_1t,$$

$$\frac{\partial H}{\partial u}=\lambda_2 e^{\frac{1}{2}\left(1-u^2\right)}\left(1-u^2\right)=0\Rightarrow u\pm 1,$$

since $\boldsymbol{\lambda}$ must be a non-trivial solution.

Next, let $signU(0)=\sigma$, which implies $\dot{y}=\sigma$, which arises from the fact that for $u=1$, or $u=-1$, then the exponential component is equal to 1. This implies that $y=\sigma t+K_1$. Applying the initial conditions for y implies that $K_1=0$. Now, returning to the x component, we have $\dot{x}=\sigma t$ which integrates to $x=\frac{\sigma t^2}{2}+K_2$. Applying the initial condition for x, we have that $K_2=0$. Thus

$$x=\frac{\sigma t^2}{2}, \qquad y=\sigma t.$$

If u does not change sign then it is not possible to satisfy the boundary/end condition, $y(T)=0$, then this implies that we need at least one switch, which at this point means that u has a discontinuity. Therefore, u must have a switch at say time $t_s\in(0,T)$, which implies that $u=-\sigma$ for $t\in[t_s,T]$ and $u=\sigma$ for $t\in[0,t_s]$.

We already know that the expressions for x and y satisfy the initial conditions for $t \in [0, t_s]$. We now consider the differential equations for x and y in the second interval, $t \in [t_s, T]$, and have them satisfy the boundary/end conditions. Therefore, $\dot{y} = -\sigma$ which implies that $y = -\sigma t + K_3$. Applying the end condition results in $y = \sigma\,(t - T)$. Next, considering the differential equation for x we have $\dot{x} = -\sigma\,(t - T)$, which implies that $x = -\frac{\sigma}{2}\,(t - T)^2 + K_4$, where after applying the end condition results, in $x - \frac{\sigma}{2}\,(t - T)^2 + 1$.

We next have to ensure the continuity of the functions when the control changes sign. Therefore, it is required that $y\,(t_s) = \sigma t_s = \sigma\,(T - t_s)$, which implies that $t_s = \frac{T}{2}$. It is also required that $x\,(t_s) = \frac{\sigma t_s^2}{2} = -\frac{\sigma}{2}\,(t_s - T)^2 + 1 \Rightarrow t_s^2 + (t_s - T)^2 = \frac{2}{\sigma}$. Substituting $t_s = \frac{T}{2}$ yields $T^2 = 4\sigma$ which implies that $\sigma = 1$ and the final time is $T = 2$. Therefore, the final solutions are

$$\begin{aligned} &t \in [0, 1], \quad u = +1, \quad x = \frac{t^2}{2}, && y = t, \\ &t \in [1, 2], \quad u = -1, \quad x = 1 - \frac{(t-2)^2}{2}, && y = 2 - t. \end{aligned} \tag{7.76}$$

The solutions for x, y and u are graphically presented in Fig. 7.6, where we can see where the control u changes sign, the component y increases linearly until $t = 1$ and then decreases linearly to $t = 2$, and finally we can see the quadratic nature of the x component, which increases quadratically until $t = 1$ and then decreases quadratically to $t = 2$.

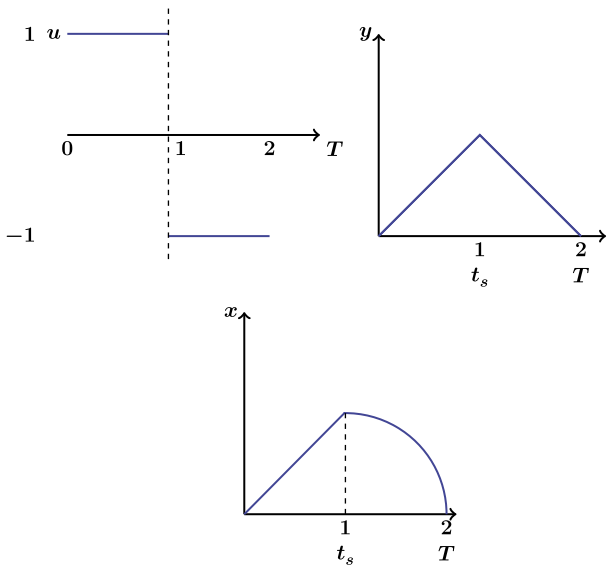


FIGURE 7.6

Plot of the optimal values of x, y, and u, respectively.

7.7 Maximization of Constrained Control Problems

Unlike in the previous sections in this chapter, we are now seeking the necessary conditions for a **strong relative maximum**. We start with the **Weierstrass Necessary Condition**: for $\boldsymbol{y}$ to give a strong relative maximum of the functional

$$\int_a^b G(\boldsymbol{y}, \dot{\boldsymbol{y}}, t)\, dt,$$

we require

$$G(\boldsymbol{y}, \boldsymbol{w}, t) - G(\boldsymbol{y}, \dot{\boldsymbol{y}}, t) - (\boldsymbol{w} - \dot{\boldsymbol{y}})^T \left.\frac{\partial G}{\partial \dot{\boldsymbol{y}}}\right|_{\boldsymbol{y}, \dot{\boldsymbol{y}}} \leq 0, \tag{7.77}$$

for all admissible $\boldsymbol{w} \in \mathbb{D}_0[a, b]$, where $\boldsymbol{y}$ and $\dot{\boldsymbol{y}}$ are optimal. We shall not prove this condition here, but we do consider the implications for the optimal control problem.

Let $\boldsymbol{y} = \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{U} \\ \boldsymbol{\lambda} \end{pmatrix}$, $\boldsymbol{U}(t) = \int_0^t \boldsymbol{u}(s)\, ds$ and $\dot{\boldsymbol{y}} = \begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{U}} \\ \dot{\boldsymbol{\lambda}} \end{pmatrix} \equiv \begin{pmatrix} \dot{\boldsymbol{x}} \\ \boldsymbol{u} \\ \dot{\boldsymbol{\lambda}} \end{pmatrix}$.

First write G in terms of the Hamiltonian $G(\boldsymbol{y}, \dot{\boldsymbol{y}}, t) = H\left(\boldsymbol{x}, \dot{\boldsymbol{U}}, t\right) - \boldsymbol{\lambda}^T \dot{\boldsymbol{x}}$. Now let $\boldsymbol{w} = \begin{pmatrix} \boldsymbol{z} \\ \boldsymbol{v} \\ \boldsymbol{\varepsilon} \end{pmatrix}$ and assume that $\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{v}, t)$.

Considering the Weierstrass condition, we require an expression for $G(\boldsymbol{y}, \boldsymbol{w}, t)$, defined by

$$G(\boldsymbol{y}, \boldsymbol{w}, t) \equiv H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) - \boldsymbol{\lambda}^T \boldsymbol{z}, \tag{7.78}$$

so that

$$\frac{\partial G}{\partial \dot{\boldsymbol{y}}} = \begin{pmatrix} \frac{\partial G}{\partial \dot{\boldsymbol{x}}} \\ \\ \frac{\partial G}{\partial \dot{\boldsymbol{U}}} \\ \\ \frac{\partial G}{\partial \dot{\boldsymbol{\lambda}}} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\lambda} \\ \\ \frac{\partial G}{\partial \boldsymbol{u}} \\ \\ \boldsymbol{0} \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\lambda} \\ \\ \boldsymbol{0} \\ \\ \boldsymbol{0} \end{pmatrix}. \tag{7.79}$$

Therefore, the Weierstrass condition for the control problem is given by

$$H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) - \boldsymbol{\lambda}^T \boldsymbol{z} - H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t) + \boldsymbol{\lambda}^T \dot{\boldsymbol{x}} - (\boldsymbol{w} - \boldsymbol{y})^T \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix} \leq 0,$$

$$\Rightarrow H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) - H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t) - \boldsymbol{\lambda}^T (\boldsymbol{z} - \dot{\boldsymbol{x}}) + (\boldsymbol{z} - \dot{\boldsymbol{x}})^T \boldsymbol{\lambda} - (\boldsymbol{v} - \boldsymbol{u})^T \frac{\partial H}{\partial \boldsymbol{u}} \leq 0, \tag{7.80}$$

but the last term in (7.80) is equal to zero, and the third and fourth terms cancel, which implies

$$H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) \leq H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t), \tag{7.81}$$

where (7.81) is the **Pontryagin maximum principle**. The interpretation of this principle is that the Hamiltonian H is **maximized** along the optimal with respect to the control $\boldsymbol{u}$.

7.7.1 Constrained Control Problems

We now consider the class of problems where the values and types of controls are bounded. We start with the functional

$$\begin{aligned} \max \int_{t_o}^{t_f} F(\boldsymbol{x}, \boldsymbol{u}, t)\, dt, \\ \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t), \quad \boldsymbol{x} \in \mathbb{R}^n, \\ \boldsymbol{x}(t_0) = \boldsymbol{x}_0, \quad \boldsymbol{u} \in \mathbb{R}^m, \\ \boldsymbol{g}\left(\boldsymbol{x}\left(t_f\right), t_f\right) = 0, \quad \boldsymbol{g} \in \mathbb{R}^p, \end{aligned}$$

where t_f may or may not be specified, over all controls $\boldsymbol{u} \in \mathbb{D}_0\left[t_0, t_f\right]$ and $\boldsymbol{u} \in \Omega$. We are going to assume that Ω is a closed convex set in $\mathbb{R}^m$. An example of a convex set would be to consider Ω to be the set such that $\alpha_i \leq u_i(t) \leq \beta_i, \forall t$, which is equivalent to defining a cube in the $\mathbb{R}^3$ space.

The necessary conditions, derived by the same arguments that we have used before, are

$$H = F(\boldsymbol{x}, \boldsymbol{u}, t) + \boldsymbol{\lambda}^T \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t),$$

where the original equations are

$$\begin{aligned} \dot{\boldsymbol{\lambda}} &= -\frac{\partial H}{\partial \boldsymbol{x}}, \\ \boldsymbol{\lambda}\left(t_f\right) &= \sum_{j=1}^{p} \mu_j \frac{\partial g_j}{\partial \boldsymbol{x}}. \end{aligned}$$

If t_f is not specified, then

$$H|_{t=t_f} = -\sum_{j=1}^{p} \mu_j \left.\frac{\partial g_j}{\partial t}\right|_{t=t_f}.$$

It is not necessary for $\frac{\partial H}{\partial \boldsymbol{u}} = \boldsymbol{0}$ since the optimal lies on the boundary of Ω. However, the maximum principle still holds, since the Weierstrass theorem implies that

$$\begin{aligned} H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) - H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t) - (\boldsymbol{v} - \boldsymbol{u})^T \frac{\partial H}{\partial \boldsymbol{u}} \leq 0, \\ \Rightarrow H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) - H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t) \leq (\boldsymbol{v} - \boldsymbol{u})^T \frac{\partial H}{\partial \boldsymbol{u}} \leq 0, \end{aligned}$$

$\forall \boldsymbol{v} \in \mathbb{D}_0$ and $\boldsymbol{v} \in \Omega$. If we consider the diagram in Fig. 7.7, then we can see that the direction, $\boldsymbol{u} - \boldsymbol{v}$, is going out of the region. Therefore $\frac{\partial H}{\partial \boldsymbol{u}}$ will be positive in the direction just given, which implies that $(\boldsymbol{u} - \boldsymbol{v}) \frac{\partial H}{\partial \boldsymbol{u}} \geq 0$, and therefore we also have that $(\boldsymbol{v} - \boldsymbol{u}) \frac{\partial H}{\partial \boldsymbol{u}} \leq 0$, and the maximum principle holds. Thus the final necessary condition is

$$H(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{\lambda}, t) \leq H(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}, t), \quad \forall \boldsymbol{v} \in \Omega,$$

along the optimal $(\boldsymbol{x}, \boldsymbol{\lambda})$.

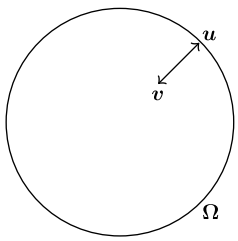


FIGURE 7.7

Plot of the difference between v and u, given the constraint Ω on u.

Example 7.13. *Consider the situation where we have a train at point $-a$ and we wish to minimize the time it takes to have the train stopped in a shed at location 0. Find the solution, and control acceleration, u, within $-\alpha \leq u \leq \beta$, and $\alpha = \beta = 1$. Consider the frictionless equation of motion $\ddot{x} = u$ and introduce the new variables x_1 and x_2 which represent the position and velocity, respectively. Therefore the system of ordinary differential equations are*

$$\dot{x}_1 = x_2,$$
$$\dot{x}_2 = u.$$

We have already considered a minimum time functional in the previous examples; therefore, for this situation we have the functional

$$\int_0^T 1 dt, \qquad \begin{matrix} x_1(0) = a, & x_1(T) = 0, \\ x_2(0) = 0, & x_2(T) = 0, \end{matrix}$$

where T is not specified.

The Hamiltonian for this problem is given by $H = -1 + \lambda_1 x_2 + \lambda_2 u$, which gives rise to the necessary conditions

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial x_1} = 0 \Rightarrow \lambda_1 = c_1,$$
$$\dot{\lambda}_2 = -\frac{\partial H}{\partial x_2} = -\lambda_1 \Rightarrow \lambda_2 = c_2 - c_1 t.$$

As T is not specified we also have the condition, $H|_T = 0$; however, the system is autonomous so the condition becomes $H = 0\ \forall t$. We also have the maximum principle which implies that H is maximized with respect to u. Therefore, we only need to consider the terms involving u in H, i.e., $H \sim \lambda_2 u \Rightarrow u = \beta = +1$, if $\lambda_2 = c_2 - c_1 t > 0$, but $u = -\alpha = -1$ if $\lambda_2 = c_2 - c_1 t < 0$. Thus the control switches from ± 1 to opposite when $\lambda_2 = 0$ and λ_2 is equal to a straight line that crosses the axis only once, thus there exists at most one switch for this problem.

To visualize this situation, we consider two cases:

Case (i) $u = +1 \Rightarrow \left.\begin{matrix} \dot{x}_1 = x_2 \\ \dot{x}_2 = 1 \end{matrix}\right\} \Rightarrow \begin{matrix} x_2 = t + S_2, \\ x_1 = \frac{1}{2}(t + S_2)^2 + S_1, \end{matrix}$ where S_1 and S_2 are arbitrary. Finally, expressing x_1 in terms of x_2 we have $\frac{1}{2}x_2^2 + S_1 = x_1$, which describes a family of parabolas, which are shown in Fig. 7.8A.

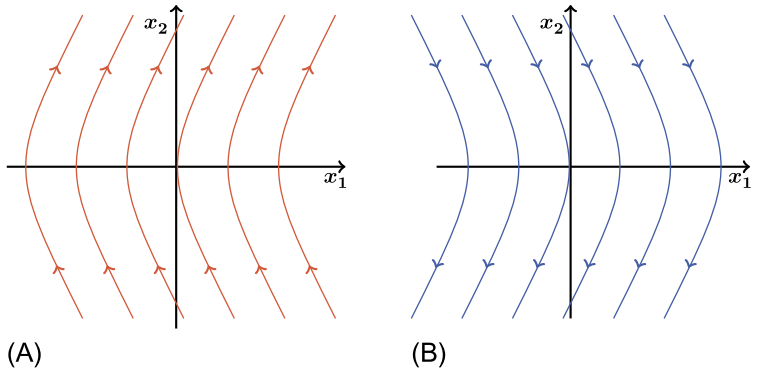


FIGURE 7.8

Phase plot of (A) $u = +1$ and (B) $u = -1$.

Case (ii) $u = -1 \Rightarrow$ $\left.\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -1 \end{aligned}\right\} \Rightarrow \begin{aligned} x_2 &= -t + \tilde{S}_2, \\ x_1 &= -\tfrac{1}{2}\left(t + \tilde{S}_2\right)^2 + \tilde{S}_1 \end{aligned}$ where $\tilde{S}_1$ and $\tilde{S}_2$ are also arbitrary.

Again finding an expression for x_1 in terms of x_2 yields $x_1 = -\frac{1}{2}x_2^2 + \tilde{S}_1$, which again describes a family of parabolas, but this time in the opposite direction to that for case (i); see Fig. 7.8B.

Therefore the optimal must lie on one of these curves from Fig. 7.8, and can switch from one to another only once, where u switches. In Fig. 7.9 we have combined the parabolas from the two plots in Fig. 7.8 and have illustrated the solution where we follow with $u = +1$ until we reach the curve into the origin corresponding to $u = -1$, where we switch at this point.

The curve AOB is called the **switching curve**. To find particular solution we have to solve the equation in the first part of the interval $[t, t_s]$ to satisfy the initial condition, and then the solve the other equations in the second interval $[t_s, T]$ to satisfy the end conditions. As with the previous example, we

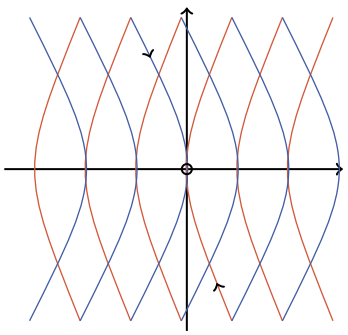


FIGURE 7.9

Phase plane plot where the optimal choice for the control is highlighted with the *black arrow*.

assume that $\sigma = u(0)$, where $\sigma = \pm 1$. We have also shown in the previous example that the switching time is at $\frac{T}{2}$, but we do not know T.

In the first interval $[0, t_s]$ we have

$$\dot{x}_2 = \sigma \Rightarrow x_2 = \sigma t + c_2 \Rightarrow x_2 = \sigma t,$$
$$\dot{x}_1 = x_2 \Rightarrow x_1 = \frac{\sigma t^2}{2} + c_1 \Rightarrow \frac{\sigma t^2}{2} - a.$$

In the second interval $[t_s, T]$ we have

$$\dot{x}_2 = -\sigma \Rightarrow x_2 = -\sigma t + \tilde{c}_2 \Rightarrow x_2 = \sigma (T - t),$$
$$\dot{x}_1 = x_2 \Rightarrow x_1 = -\frac{\sigma (t - T)^2}{2} + \tilde{c}_1 \Rightarrow x_1 = -\frac{\sigma}{2} (t - T)^2.$$

Finally, we need to match the two sets of the solutions at $t = t_s$:

$$x_2(t_s) = \sigma t_s = \sigma (T - t_s) \Rightarrow t_s = \frac{T}{2}, \tag{7.82}$$
$$x_1(t_s) = \frac{\sigma t_s^2}{2} - a = -\frac{\sigma (t - T)^2}{2} \Rightarrow \sigma t_s^2 = a. \tag{7.83}$$

We require $\sigma > 0$ since a is positive, therefore $\sigma = +1 \Rightarrow u(0) = +1$:

$$t_s = \sqrt{a} \Rightarrow T = s\sqrt{a}.$$

Therefore, the control $u = +1$ for $t \in \left[0, \sqrt{a}\right]$, and $u = -1$ for $t \in \left[\sqrt{a}, 2\sqrt{a}\right]$.

To help illustrate the constrained problems, we return to the **oscillating problem**:

$$\min \int_0^T -1 dt,$$
$$\begin{aligned} &\dot{x}_1 = x_2, && x_1(0) = x_0, && x_1(T) = 0, \\ &\dot{x}_2 = -x_1 + u, && x_2(0) = v_0, && x_2(T) = 0, \\ &|u| \leq 1. \end{aligned}$$

Forming the Hamiltonian, we have $H = -1 + \lambda_1 x_2 + \lambda_2 (u - x_1)$, where the associated adjoint equations are

$$\dot{\lambda}_1 = -\frac{\partial H}{\partial x_1} = \lambda_2,$$
$$\dot{\lambda}_2 = -\frac{\partial H}{\partial x_2} = -\lambda_1.$$

We now need to maximize the Hamiltonian with respect to the control u. Therefore, we have to maximize $\lambda_2 u$, which implies that $u = +1$ for $\lambda_2 > 0$ and $u = -1$ for $\lambda_2 < 0$.

We need to combine the two differential equations for $\lambda_{1,2}$ through

$$\ddot{\lambda_2} = -\dot{\lambda_1} = -\lambda_2 \Rightarrow \lambda_2 = A \cos t + b \sin t \equiv \tilde{A} \sin (t + \gamma),$$

which implies that λ_2 changes sign periodically with a period of 2π, and that the control u switches between $+1$ and -1 at intervals of π. Thus

$$\ddot{x}_1 = \dot{x}_2 = -x_2 \pm 1. \tag{7.84}$$

- If $u = +1$ then $\begin{aligned} x_1 &= R\cos(t+\beta)+1 \\ x_2 &= -R\sin(t+\beta) \end{aligned}$,
- if $u = -1$ then $\begin{aligned} x_1 &= R\cos(t+\beta)-1 \\ x_2 &= -R\sin(t+\beta) \end{aligned}$.

We choose R, β, $\tilde{A}$, γ, T to satisfy the initial conditions, end conditions, and $H|_T = 0$.

Rearranging the two expressions above by moving the ± 1 over the left-hand sides for the two x_1 expressions and forming the square of both sides and finally adding the square of x_2 results in

$$(x_1 - 1)^2 + x_2 = R^2, \qquad u = +1, \tag{7.85}$$

$$(x_1 + 1)^2 + x_2 = R^2, \qquad u = -1, \tag{7.86}$$

which implies that the phase plane consists of circles. We have plotted the phase plane for this situation and have highlighted a possible optimal solution, as we have not solved for R yet, in Fig. 7.10.

Therefore, to interpret Fig. 7.10, we see that if we are below the red curve then we take $u = +1$, and switch each time the solution hits the recurve; otherwise we take $u = -1$ if above the red curve.

7.8 Two Classical Optimal Control Problems

The two types of problems that we consider in this section are:

1. **linear quadratic regulator (LQR) problems**, time specified; and
2. **time optimal problems**, Pontryagin theory,

for linear time-invariant (LTI) systems.

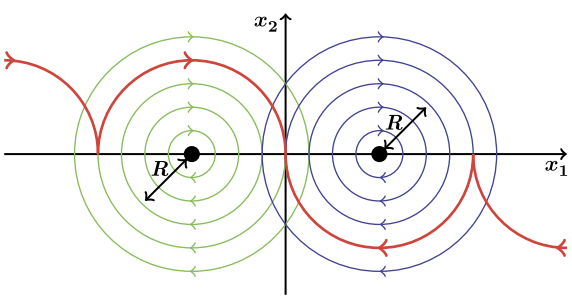


FIGURE 7.10

Phase plane plot where the optimal choice for the control is highlighted in *red*.

We start by recalling that the general control system problem is defined by

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u},\ \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{7.87}$$

where $\mathbf{A}$ and $\mathbf{B}$ are constant matrices, and $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, $t \in [0, T]$.

The first property that we consider for the linear time-invariant system is controllability, which is defined as follows.

Definition 7.14. Controllability: The system is completely controllable if and only if for every $\boldsymbol{x}_0$, and $\boldsymbol{x}_T$, there exists an admissible control that transfers the system from $\boldsymbol{x}(0) = \boldsymbol{x}_0$ at $t = 0$ to $\boldsymbol{x}(T) = \boldsymbol{x}_T$ at $t = T$ for some finite value to T.

The next definition that we recall here is stability.

Definition 7.15. Stability—asymptotic stability: The system $\dot{\boldsymbol{x}} = \mathbf{A}$ is in equilibrium at $\boldsymbol{x} = \mathbf{0}$, with $\dot{\boldsymbol{x}} = \mathbf{0}$. If we perturb the initial conditions by $\boldsymbol{\delta}$, then the system is asymptotically stable if the response, the solution of the differential equation, converges to zero.

Finally we recall the stability theorem from Chapter 6 where the linear time-invariant system is asymptotically stable if and only if the real component of the eigenvalues of $\mathbf{A}$, λ_i are less than zero.

Starting with the linear quadratic regulator problem, the associated functional problem is defined by

$$J = \min \frac{1}{2}\int_0^T \boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} + \boldsymbol{u}^T\mathbf{Q}\boldsymbol{u}dt \equiv \max -\frac{1}{2}\int_0^T \boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} + \boldsymbol{u}^T\mathbf{Q}\boldsymbol{u}dt, \tag{7.88}$$

where it is assumed that $\mathbf{Q}$ is semipositive definite, or non-negative definite, that is to say $\boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} \geq 0$, $\forall \boldsymbol{x} \neq \mathbf{0}$, $\boldsymbol{x} \in \mathbb{R}^n$. We shall assume $\mathbf{R}$ is strictly positive definite $\boldsymbol{u}^T\mathbf{R}\boldsymbol{u} > 0$, $\forall \boldsymbol{u} \neq \mathbf{0}$, $\boldsymbol{u} \in \mathbb{R}^m$. The final piece of the linear quadratic regulator problem is to constrain (7.88) by (7.87).

Forming the necessary conditions, we require the associated Hamiltonian

$$H = -\frac{1}{2}\boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} - \frac{1}{2}\boldsymbol{u}^T\mathbf{R}\boldsymbol{u} + \boldsymbol{\lambda}^T(\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}), \tag{7.89}$$

and the following derivatives:

$$\frac{\partial H}{\partial \boldsymbol{u}} = \mathbf{0} \Rightarrow -\mathbf{R}\boldsymbol{u} + \mathbf{B}^T\boldsymbol{\lambda} \Rightarrow \boldsymbol{u} = \mathbf{R}^{-1}\mathbf{B}^T\boldsymbol{\lambda}, \tag{7.90}$$

$$\dot{\boldsymbol{\lambda}} = -\frac{\partial H}{\partial \boldsymbol{x}} = -\mathbf{A}^T\boldsymbol{\lambda} + \mathbf{Q}\boldsymbol{x}. \tag{7.91}$$

Combining (7.90) and (7.91) into (7.87) results in the following $2n$-dimensional linear time-invariant system

$$\begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\lambda}} \end{pmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T \\ \mathbf{Q} & -\mathbf{A}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\lambda} \end{bmatrix}. \tag{7.92}$$

As $x(T)$ is not specified, we need to satisfy $\boldsymbol{\lambda}(T) = 0$. Therefore, we have a $2n$ linear time-invariant dimensional system with $2n$ boundary conditions $\begin{cases} \boldsymbol{x}(0) = \boldsymbol{x}_0, \\ \boldsymbol{\lambda}(T) = \mathbf{0}. \end{cases}$

We are therefore going to form a feedback solution where we have $\boldsymbol{u}$ in terms of $\boldsymbol{\lambda}$; we look for a solution for $\boldsymbol{\lambda}$ of the form $\boldsymbol{\lambda} = \mathbf{K}\boldsymbol{x}$. We assume that $\dot{\boldsymbol{\lambda}} = \mathbf{K}\dot{\boldsymbol{x}} + \dot{\mathbf{K}}\boldsymbol{x}$. Therefore, substituting for $\dot{\boldsymbol{\lambda}}$ and $\dot{\boldsymbol{x}}$

implies that

$$\mathbf{K}\left(\mathbf{A}\boldsymbol{x}+\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\boldsymbol{\lambda}\right)+\dot{\mathbf{K}}\boldsymbol{x}-\mathbf{Q}\boldsymbol{x}+\mathbf{A}^{T}\boldsymbol{\lambda}=\mathbf{0}. \tag{7.93}$$

Substituting for $\boldsymbol{\lambda}$ in (7.93) results in

$$\begin{aligned}&\mathbf{K}\left(\mathbf{A}\boldsymbol{x}+\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}\right)\boldsymbol{x}+\dot{\mathbf{K}}\boldsymbol{x}-\mathbf{Q}\boldsymbol{x}+\mathbf{A}^{T}\mathbf{K}\boldsymbol{x}=\mathbf{0},\\ &\Rightarrow\left(\mathbf{K}\mathbf{A}+\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}+\mathbf{A}^{T}\mathbf{K}-\mathbf{Q}+\dot{\mathbf{K}}\right)\boldsymbol{x}=\mathbf{0}.\end{aligned} \tag{7.94}$$

For all values of t and for any choice of $\boldsymbol{x}(0)$, we require $\mathbf{K}$ to satisfy the matrix **Riccati differential equation**:

$$\dot{\mathbf{K}}+\mathbf{K}\mathbf{A}+\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}+\mathbf{A}^{T}\mathbf{K}-\mathbf{Q}=\mathbf{0}, \tag{7.95}$$

and since $\boldsymbol{\lambda}(T)=\mathbf{0}$, this implies that $\mathbf{K}\boldsymbol{x}(T)=\mathbf{0}$, but this has to hold for all possible values of $\boldsymbol{x}(T)$, therefore this implies that $\mathbf{K}(T)=\mathbf{0}$.

We now assume that $\mathbf{K}$ is a symmetric matrix, and we therefore seek a symmetric solution for $\mathbf{K}$. Thus, solving for $\mathbf{K}$ yields

$$\boldsymbol{u}=\mathbf{R}^{-1}\mathbf{B}^{T}\boldsymbol{\lambda}\equiv\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}\boldsymbol{x}, \tag{7.96}$$

where the expression $\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}$ is referred to as the **feedback gain**.

Therefore, the closed loop linear time-invariant system is given by

$$\dot{\boldsymbol{x}}=\left(\mathbf{A}+\mathbf{B}\mathbf{R}^{-1}\mathbf{B}\mathbf{K}\right)\boldsymbol{x}. \tag{7.97}$$

Given the derivation above, we are now able to prove the following property of the cost function, J:

$$\min J=-\frac{1}{2}\boldsymbol{x}_{0}^{T}\mathbf{K}(0)\boldsymbol{x}_{0}. \tag{7.98}$$

Proof.

$$\begin{aligned}\int_{0}^{T}\frac{d}{dt}\left(\boldsymbol{x}^{T}\mathbf{K}\boldsymbol{x}\right)dt&=\underbrace{\boldsymbol{x}(T)^{T}\mathbf{K}(T)\boldsymbol{x}(T)}_{=0,\,(\mathbf{K}(T))=0}-\boldsymbol{x}(0)^{T}\mathbf{K}(0)\boldsymbol{x}(0),\\ \Rightarrow-\frac{1}{2}\boldsymbol{x}_{0}^{T}\mathbf{K}(0)\boldsymbol{x}_{0}&=\frac{1}{2}\int_{0}^{T}\frac{d}{dt}\left(\boldsymbol{x}^{T}\mathbf{K}\boldsymbol{x}\right)dt,\\ &=\frac{1}{2}\int_{0}^{T}\dot{\boldsymbol{x}}^{T}\mathbf{K}\boldsymbol{x}+\boldsymbol{x}^{T}\dot{\mathbf{K}}\boldsymbol{x}+\boldsymbol{x}^{T}\mathbf{K}\dot{\boldsymbol{x}}dt,\\ &=\frac{1}{2}\int_{0}^{T}\boldsymbol{x}^{T}\left\{\left(\mathbf{A}^{T}\mathbf{K}+\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}+\dot{\mathbf{K}}+\mathbf{K}\mathbf{A}\right)+\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}\right\}\boldsymbol{x}dt,\\ &=\frac{1}{2}\int_{0}^{T}\boldsymbol{x}^{T}\mathbf{Q}\boldsymbol{x}+\boldsymbol{x}^{T}\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}\boldsymbol{x}dt,\\ &=\frac{1}{2}\int_{0}^{T}\boldsymbol{x}^{T}\mathbf{Q}\boldsymbol{x}+\boldsymbol{x}^{T}\left(\mathbf{K}\mathbf{B}\mathbf{R}^{-1}\right)\mathbf{R}\left(\mathbf{R}^{-1}\mathbf{B}^{T}\mathbf{K}\right)\boldsymbol{x}dt,\end{aligned}$$

$$= \frac{1}{2}\int_0^T \boldsymbol{x}^T \mathbf{Q}\boldsymbol{x} + \boldsymbol{u}^T \mathbf{R}\boldsymbol{u} dt,$$

where we have used the common trick $\mathbf{I} = \mathbf{R}\mathbf{R}^{-1}$.

Example 7.16. $\min \int_0^1 x^2 + \frac{u^2}{3} dt \equiv \frac{1}{2}\int_0^1 2x^2 + \frac{2}{3}u^2 dt$ *subject to* $\dot{x} = u - x$ *and* $x(0) = x_0$.

The first step is to form $F = \frac{1}{2}\left(\boldsymbol{x}^T \mathbf{Q}\boldsymbol{x} + \boldsymbol{u}^T \mathbf{R}\boldsymbol{u}\right)$, which for this example we have $Q = 2$, $R = \frac{2}{3}$, $A = -1$, $B = 1$. Next, we require the Hamiltonian, which is $H = -\left(x^2 + \frac{1}{3}u^2\right) + \lambda(u - x)$, $\frac{\partial H}{\partial u} = 0 \Rightarrow u = \frac{3}{2}\lambda$. Now forming the Riccati equation, we have

$$\begin{aligned}\dot{K} &= -\frac{3}{2}K^2 + 2K + 2,\ K(1) = 0,\\ &= (2-k)\left(\frac{3}{2}K + 1\right) = \frac{1}{2}(2-K)(3K+2).\end{aligned}$$

Applying separation of variables and partial fractions leads to

$$2\int_1^t \frac{dK}{(2-K)(3K+2)} = \int_1^t dt.$$

We next solve for the constants in the partial fractions, which can be shown to be

$$\frac{1}{(2-K)(3K+2)} = \frac{\tilde{A}}{(2-K)} + \frac{\tilde{B}}{(3K+2)} \Rightarrow \tilde{A} = \frac{1}{8},\ \tilde{B} = \frac{3}{8}.$$

Therefore,

$$\begin{aligned}&\int_1^t -\frac{1}{4}\frac{(-dK)}{2-K} + \int_1^t \frac{1}{4}\frac{3dK}{(3K+2)} = \int_1^t dt,\\ &= -\frac{1}{4}\log(2-K)\Big|_1^t + \frac{1}{4}\log(3K+2)\Big|_1^t = t-1,\\ &4(t-1) = \log\frac{3K+2}{2-K} \Rightarrow 3K+2 = (2-K)e^{4t-1},\\ &K = \frac{2\left(e^{4(t-1)} - 1\right)}{3 + e^{4(t-1)}},\end{aligned}$$

which is the feedback. The optimal control is $u = \frac{3}{2}\lambda = \frac{3}{2}Kx$. The optimal state satisfies

$$\dot{x} = \left(-1 + \frac{3}{2}K\right)x, \quad x(0) = 1,$$

which can be shown to integrate to

$$x = \hat{A}\frac{3 + e^{4(t-1)}}{e^{2(t-1)}} \Rightarrow \hat{A} = \frac{e^{-2}}{e^{-4} + 3}. \tag{7.99}$$

Thus the control u is

$$u = \frac{3\sinh(2(t-1))}{\sinh 2 + 2\cosh 2}, \tag{7.100}$$

and the value of the minimum of the cost function is

$$J_{min} = -\frac{1}{2}x_0 K(0)\, x_0 = -\frac{1}{2}K(0)\, x(0)^2,$$
$$= -\frac{e^{-4}-1}{3+e^{-4}} = \frac{1-e^{-4}}{3+e^{-4}}.$$

We now consider a special case where we allow T to tend to infinity. Therefore we have the functional problem

$$\min \frac{1}{2}\int_0^{\infty} \boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} + \boldsymbol{u}^T\mathbf{R}\boldsymbol{u}dt,$$
$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u},\ \boldsymbol{x}(0) = \boldsymbol{x}_0.$$

This then implies that the optimal control is $\boldsymbol{u} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{K}\boldsymbol{x}$, where $\mathbf{K}$ satisfies the *steady state* Riccati differential equation, which is referred to as the **algebraic Riccati equation**:

$$\mathbf{KA} + \mathbf{A}^T\mathbf{K} + \mathbf{KBR}^{-1}\mathbf{B}^T\mathbf{K} = \mathbf{Q}. \tag{7.101}$$

It is important to note here that there exists a unique symmetric positive definite solution of the algebraic Riccati equation in (7.101), and it has the important property that it guarantees that

$$\mathbf{A} + \mathbf{BR}^{-1}\mathbf{B}^T\mathbf{K},$$

is **asymptotically stable**.

For the linear quadratic regulator problem, in general, there exists a unique solution for the necessary conditions, assuming that the system is well posed.

The solution to the necessary conditions, the extremals, result in a relative stationary value for the cost function J, in general. It is also possible to show that if $\boldsymbol{x}^*$, $\boldsymbol{\lambda}^*$ and $\boldsymbol{u}^*$ give the extremal satisfying the necessary conditions, then

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{\lambda}) = \frac{1}{2}\int_{t_0}^{T} -\boldsymbol{x}^T\mathbf{Q}\boldsymbol{x} - \boldsymbol{u}^T\mathbf{R}\boldsymbol{u} + \boldsymbol{\lambda}(\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u} - \dot{\boldsymbol{x}})\, dt \leq \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{\lambda}^*), \tag{7.102}$$

implies that the solution of the necessary conditions exist, if the system is completely controllable, and is a unique global minimum.

To finish off this chapter we now address the properties of the Pontryagin-type problems, which we recall mathematically are defined as

$$\min \int_0^T dt,$$

where the controls satisfy $\alpha_i \leq u_i \leq \beta_i$, as well as the end condition $\boldsymbol{x}(T) = \boldsymbol{x}_f = \mathbf{0}$, subject to the control system $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}$, combined with the initial conditions $\boldsymbol{x}(0) = \boldsymbol{x}_0$, and are dependent on the control system properties of controllability and stability.

The necessary conditions for a solution to these types of problems are found through maximizing the Hamiltonian

$$H = -1 + \boldsymbol{\lambda}^T (\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}), \tag{7.103}$$

with respect to $\boldsymbol{u}$. Thus, we have to find the maximum of $\boldsymbol{\lambda}^T \mathbf{B}\boldsymbol{u}$.

Note:

$$\boldsymbol{\lambda}^T \mathbf{B}\boldsymbol{u} = \boldsymbol{\lambda}^T \begin{bmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \dots & \boldsymbol{b}_m \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix} = \sum_{i=1}^{m} \boldsymbol{\lambda}^T \boldsymbol{b}_i u_i. \tag{7.104}$$

This implies that

- $u_i = \beta_i$ if $\boldsymbol{\lambda}^T \boldsymbol{b}_i = S_i(t) > 0$; and
- $u_i = \alpha_i$ if $\boldsymbol{\lambda}^T \boldsymbol{b}_i = S_i(t) < 0$,

where $S_i(t)$ is a switching function and is equal to zero for $t \in [t_1, t_2]$. This is to say that on some interval, a control u_i is undefined there. We now address the question of whether or not this can happen.

As we do not have a specified end time, we require the extra condition $H|_{t=T} = 0$. Therefore, we require $H = -1 + \boldsymbol{\lambda}^T (\mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u}) = 0$, and must be true for all t, since the system is autonomous, which implies that $\boldsymbol{\lambda}(t) \neq \mathbf{0}$ for any $t \in (t, T)$. We also have that $s_i = 0$ only if $\boldsymbol{\lambda}^T \boldsymbol{b}_i = 0$ on some interval. Differentiating the switch function definition, we have $\dot{\boldsymbol{\lambda}} \boldsymbol{b}_i = -\boldsymbol{\lambda}^T \mathbf{A}\boldsymbol{b}_i = 0$. We repeat this step to form an induction proof, which implies $\boldsymbol{\lambda}^T \mathbf{A}^j \boldsymbol{b}_i$ for $j = 1, 2, \dots, n-1$. Therefore we have

$$\boldsymbol{\lambda}^T \begin{bmatrix} \boldsymbol{b}_i & \mathbf{A}\boldsymbol{b}_i & \dots & \mathbf{A}^j \boldsymbol{b}_i \end{bmatrix} = 0,$$

on some interval $[t_1, t_2]$, which implies $\begin{bmatrix} \boldsymbol{b}_i & \mathbf{A}\boldsymbol{b}_i & \dots & \mathbf{A}^j \boldsymbol{b}_i \end{bmatrix}$ is full rank. Thus, the only solution is $\boldsymbol{\lambda} = \mathbf{0}$ on some interval, which is a contradiction. Therefore, we have that $s_i \neq 0$ on any interval.

The significance of this result is that if the system is **completely controllable** by each of the controls, where the completely controllability property implies that the controllability matrix $\begin{bmatrix} \boldsymbol{b}_i & \mathbf{A}\boldsymbol{b}_i & \mathbf{A}^2\boldsymbol{b}_i & \dots & \mathbf{A}^j \boldsymbol{b}_i \end{bmatrix}$ has full rank, then there are no intervals over which $s_i(t) = 0$, for any i. Therefore, the necessary conditions completely determine the controls.

The controls for these types of problems are known to be piecewise constant with switches at points where $s_i(t) = 0$, are of the form $u_i = \beta_i$, $s_i > 0$, and $u_i = \alpha_i$ for $s_i < 0$, where $s_i(t) = \boldsymbol{\lambda}^T \boldsymbol{b}_i$. Therefore, the switches occur when $\boldsymbol{\lambda}$ goes through zero, but since $\dot{\boldsymbol{\lambda}} = -\mathbf{A}\boldsymbol{\lambda} \Rightarrow \boldsymbol{\lambda}(t) = e^{-\mathbf{A}t}\boldsymbol{\lambda}(0)$, this implies that λ is some linear combination of the $e^{\mu_j t}\mathbf{v}_j$, where μ_j is an eigenvalue of $-\mathbf{A}^T$, and $\mathbf{v}_j$ are the associated eigenvectors.

If μ_j are real then $e^{\mu_j t}$ are monotonic and cross the axis at most once, which implies that there are at most $n-1$ switches. If, however, the eigenvalues are complex, then it possible to have switches at periodic intervals. We have seen examples of both of these situations. For the monotonic case we have the train example, while the spring example demonstrates the complex number case.

Therefore, in conclusion, if the control system is completely controllable for each control and if $\mathbf{A}$ is stable, then it is possible to move the system from any point to the origin in finite time using an admissible control.

7.9 Summary

In this chapter we have introduced the necessary conditions to obtain the optimal control for a given cost function, subject to a control problem. We have derived the necessary conditions for when the end time is not specified, through introducing the Hamiltonian; and it is through the Hamiltonian that we can find conditions for the optimal control as a result of the equivalent Euler equations. This system of Euler equations contains an adjoint equation. Also through the Hamiltonian equation it is possible to derive the necessary conditions for when the end condition is a function and is also not dependent on a fixed end time. We extended the calculus of variation theory from Chapter 5 for several different situations that had not been covered in that chapter, and applied that theory to the optimal control problems. We have also introduced two types of problems: linear quadratic regulator and the Pontryagin maximum principle. We have also derived the differential and algebraic Riccati equations, which provide necessary conditions for the original control system to be completely controllable and asymptotically stable.

We have also presented Weierstrass corner conditions to enable us to find the optimal control when the function that represents the optimal control has jump discontinuities, where the corner condition uses the Du Bois-Reymond lemma.

The reason for introducing the control and optimal control theory over the last two chapters is to lay the grounds for linking the variational and Kalman filter data assimilation systems back to these theoretical fields.

We now move on to a very important component of data assimilation: numerical modeling. Over the next six chapters, we shall introduce different forms of numerical modeling for different types of differential equations for different grids and shapes. We start with numerical approximations to initial value problems.

CHAPTER

Numerical Solutions to Initial Value Problems

8

Contents

Differential equations play a vital part in many form of modeling and approximations to a geophysical flow or static relationships. However, many differential equations are too hard to solve analytically, and as such we form numerical approximations to find some type of solution that is assumed to closely approximate the true solution. In the control theory and the optimal control chapters we considered differential equations where it was possible to find analytical solutions. One of these differential equations was

$$\frac{dy}{dt} = \lambda y, \quad y(0) = 1, \tag{8.1}$$

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00012-7

which has the solution $y = e^{\lambda t}$ with the different characteristics shown in Fig. 8.1. We can see that if $\lambda > 0$, then the solution are growing, while if $\lambda = 0$, then the solution is constant, and finally if $\lambda < 0$, then the solution is decaying.

An equivalent numerical approximation to (8.1) could be

$$\frac{y^{n+1} - y^n}{h} = \lambda y^n, \tag{8.2}$$

where (8.2) is referred to as a **difference equation**, which defines a sequence, and the problem is to find y_n.

Now if we consider a second-order ordinary differential equation that is a function of space, and not time, of the form

$$\frac{d^2y}{dx^2} = \lambda y, \qquad y(0) = 1, \quad y'(0) = 1 \ , \tag{8.3}$$

then an associated difference equation could be of the form

$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = \lambda y_i, \qquad y_0 = 1, \quad \frac{y_N - y_{N-1}}{h} = 1, \tag{8.4}$$

where h in the difference equations (8.2) and (8.4) represent δt and δx, respectively, where the deltas refer to the change in time or space between the points in the numerical grid. We shall explain these terms as we progress through this chapter. Also note that we have used subscripts to represent changes in spatial coordinates, and superscripts to refer to the temporal coordinate.

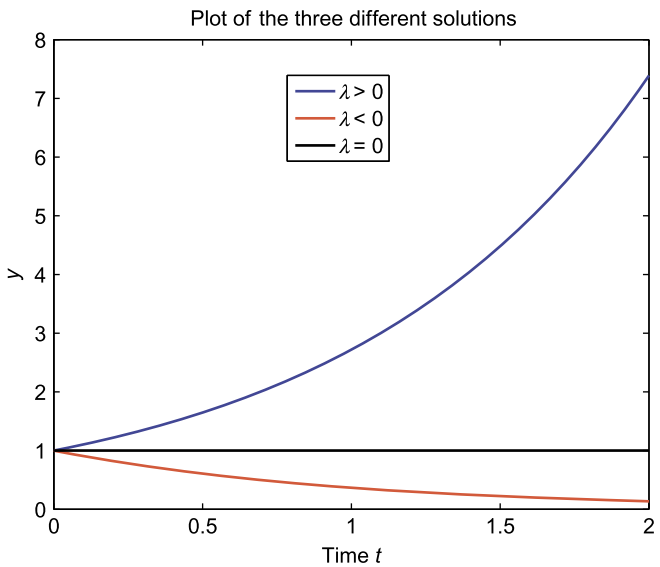


FIGURE 8.1

Plot of the three possible solutions for $y = e^{\lambda t}$.

The difference equations are a finite approximation to a continuous function. The reason being is that if we had a infinite number of points to approximate a function, then this would become difficult to store computationally. However, if we have a reduced number of points, then these are computationally storable. The mathematics associated with these two situations are **analysis and calculus**, and **algebra and sequences**, respectively, and it is from these fields that we shall draw to define and quantify the numerical approximations to the differential equations.

When approximating differential equations, we can use analysis to quantify the questions of **existence, uniqueness, and well-posedness**, while with the numerical approximation we are addressing the questions of **accuracy, numerical stability, and ill-conditioning**.

8.1 Local and Truncation Errors

If we consider the difference equation in (8.2), then we can rewrite this as

$$y^{n+1} = (1 + \lambda h)\, y^n, \tag{8.5}$$

where expanding (8.5) as a sequence starting at 0 we have

$$\begin{aligned} y^1 &= (1 + \lambda h)\, y^0, \\ y^2 &= (1 + \lambda h)\, y^1 \equiv (1 + \lambda h)\,(1 + \lambda h)\, y^0 = (1 + \lambda h)^2\, y^0, \\ &\vdots \\ y^n &= (1 + \lambda h)\, y^{n-1} \equiv (1 + \lambda h)\,(1 + \lambda h)\, y^{n-2} = \cdots = (1 + \lambda h)^n\, y_0. \end{aligned} \tag{8.6}$$

However, if $|\lambda h| > 2$ then the function defined in (8.6) will oscillate. The aim of the numerical approximation is to satisfy the condition that if we take the limit as $h \to 0$ then the approximation in (8.6) tends to the solution of (8.1).

The equation defined in (8.1) is a linear scalar problem and has the analytical solution $e^{\lambda t}$. The difference equation given in (8.2) is referred to as **Euler's method**. However, we have to evaluate how accurate the numerical approximation is, and test to see if the scheme is stable.

When we consider the accuracy of the numerical scheme, we do so by considering three types of errors: local error, truncation error, and/or global error. If we consider the Euler method, we have that $e^{\lambda t_{n+1}} = e^{\lambda (t_n + h)} = e^{\lambda h} e^{\lambda t_n}$. If we now consider the difference between the true solution $y\,(t_{n+1})$ and the numerical approximation valid at the same time, y^{n+1}, we have

$$\begin{aligned} y\left(t_{n+1}\right) - y^{n+1} &= e^{\lambda t_{n+1}} - (1 + \lambda h)\, y^n, \\ &= e^{\lambda h} y^n - (1 + \lambda h)\, y^n, \\ &= \left(e^{\lambda h} - (1 + \lambda h)\right) y^n. \end{aligned} \tag{8.7}$$

Taking the Taylor series expansion of $e^{\lambda h}$, we see that the first two terms of the Taylor series cancel with the two terms from the numerical approximation. Therefore, the error caused by this approximation is $\frac{(\lambda h)^2}{2!}$. This is the **local error**.

We next consider how to define the truncation error: define $L(y) \equiv \frac{dy}{dt} - \lambda y = 0$ and $L_h(y_n) \equiv \frac{y^{n+1}-y^n}{h} - \lambda y^n = 0$. Then, substituting the alternative errors into the two error definitions raises the question: how small is $L(y^n)$ and how small is $L_h(y)$? This is the definition of **truncation errors**.

Finally, the global error is given by $y(t_n) - y_n$. This is the most desirable error but more often than not, it is not available.

Returning to the second-order ordinary differential equation given in (8.3), we have three different types of solutions depending on whether λ is greater than, equal to, or less than zero. The general solutions to these three problems are as follows:

$$\begin{aligned}
&\lambda > 0, \quad y(x) = c_1 x + c_2, \\
&\lambda = 0, \quad y(x) = c_1 e^{\sqrt{\lambda}x} + c_2 e^{-\sqrt{\lambda}x}, \\
&\lambda < 0, \quad y(x) = c_1 \cos\sqrt{-\lambda}x + c_2 \sin\sqrt{-\lambda}x.
\end{aligned}$$

The solution for the $\lambda < 0$ case depends upon the condition at $y(0)$. There could be an infinite number of solutions, no solution, or a unique solution.

If we consider the general form of the first-order differential equation as

$$\frac{dy}{dt} = f(t, y), \quad y(0) = y_0, \tag{8.8}$$

then does the differential equation in (8.8) have a solution? If so, is it unique? The answer depends upon f. To address these questions, we consider the **Lipschitz condition**.

Definition 8.1. Lipschitz condition: If y and z are any two values, then there exists a number L such that

$$\left| \frac{f(t, y) - f(t, z)}{y - z} \right| \leq L. \tag{8.9}$$

If we have the case that f is differentiable, then it satisfies the condition in (8.9).

The definition of the Euler method $\frac{y^{n+1}-y^n}{h} = f^n$, where $f^n \equiv f(t_n, y^n)$ suffers from poor accuracy. To improve the accuracy, we can halve the step size h, which doubles the accuracy of the solution but requires more time and storage. We now consider two different approaches for increasing the accuracy of the numerical approximation:

1. linear multistep methods; and
2. Runge-Kutta methods.

The approximation $\frac{y^{n+1}-y^n}{h}$ is a crude approximation to $\frac{dy}{dt}$, which uses the approximation to the gradient. Another approach would be to consider the integration of (8.1) as

$$\int_{t^n}^{t^{n+1}} \frac{dy}{dt} dt = \int_{t^n}^{t^{n+1}} f(t, y)\, dt. \tag{8.10}$$

Analytically evaluating the left-hand side of (8.10) and then applying a quadrature rule to the right-hand sides results in

$$y^{n+1} - y^n = \frac{1}{2}h\left[f(t_n, y^n) + f\left(t_{n+1}, y^{n+1}\right)\right],$$

$$\frac{y^{n+1}-y^n}{h}=\frac{1}{2}f^n+f^{n+1}. \tag{8.11}$$

This is an example of a linear multistep.

8.2 Linear Multistep Methods

The general form of a linear multistep method is given as follows.

$$\frac{\alpha_k y^{n+k}+\alpha_{k-1}y^{n+k-1}+\cdots+\alpha_0 y^n}{h}=\beta_k f^{n+k}+\beta_{k-1}f^{n+k-1}+\cdots+\beta_0 f^n. \tag{8.12}$$

We now consider different values for k to see what the schemes become:

- $k=1$

$$\frac{\alpha_1 y^{n+1}+\alpha_0 y^n}{h}=\beta_1 f^{n+1}+\beta_0 f^n,$$

where we can have

$$\alpha_1=1,\alpha_0=-1, \qquad \beta_1=0,\beta_0=1,$$
$$\alpha_1=1,\alpha_0=-1, \qquad \beta_1=\frac{1}{2},\beta_0=-\frac{1}{2},$$

which results in the following difference equations

$$\frac{y^{n+1}-y^n}{h}=f^n,$$
$$\frac{y^{n+1}-y^n}{h}=\frac{1}{2}\left(f^{n+1}-f^n\right).$$

- $k=2$

$$\alpha_2=1,\quad \alpha_1=0,\quad \alpha_0=-1,$$
$$\beta_2=0,\quad \beta_1=2,\quad \beta_0=0,$$

which results in a **mid-point** approximation

$$\frac{y^{n+2}-y^n}{2h}=f^{n+1}.$$

Therefore, the question here is: how do we choose the values for the αs and βs so that we obtain the maximum accuracy to avoid instability? We address this question by using the **truncation error**, which is a measure of accuracy.

We now consider the following three schemes:

$$\frac{y^{n+1}-y^n}{h}=f^{n+1}, \tag{8.13a}$$

$$\frac{y^{n+1}-y^n}{h}=\frac{1}{2}\left(f^n+f^{n+1}\right), \tag{8.13b}$$

$$\frac{y^{n+2}-y^n}{h}=f^{n+1}, \tag{8.13c}$$

and for each approximate solution of $y'=f\,(t,y)$, we check the consistency conditions, determine the order of accuracy, and find a bound on the truncation error in terms of a bound on the modulus of a derivative of y.

We start by considering the general expansion of the truncation error given by

$$\tau_n=\frac{C_0}{h}+C_1y+C_2hy'+C_3h^2y''+\cdots. \tag{8.14}$$

We can see that the truncation error, τ_n, can only tend to zero if C_0 and C_1 are equal to zero, and as we know, this is the consistency condition. If we now have $C_2=C_3=C_4=0$, up to $C_p=0$ but $C_{p+1}\neq 0$, then

$$\begin{aligned}\tau_n&=C_{p+1}h^py^{(p+1)}+C_{p+2}h^{p+1}y^{(p+2)}+\cdots\\&=C_{p+1}h^py^{(p+1)}+O\left(h^{p+1}\right).\end{aligned} \tag{8.15}$$

Therefore, the first remaining value of the power of the h, in this case h^p defines the **order of accuracy**. We should recall that the p-coefficient term is defined as

$$C_p=\sum_{i=1}^{k}\left(\frac{i^p}{p!}\alpha_i-\frac{i^{(p-1)}}{(p-1)!}\beta_i\right). \tag{8.16}$$

Finally, we address how to determine the **bound on the truncation error**. We start with the Taylor series expansion of $y\,(t_n+h)$ with a remainder term as

$$y\,(t_n+h)=y\,(t_n)+\frac{h^p}{p!}y^{(p)}\,(t_n+\theta h)\,,\quad (0<\theta<1)\,. \tag{8.17}$$

Expanding the truncation error term yields

$$\tau_n=C_{p+1}h^py^{(p+1)}=\sum_{i=0}^{k}\left(\frac{i^{p+1}}{(p+1)!}\alpha_iy^{p+1}\,(\xi_i)-\frac{i^p}{p!}\beta_iy^{p+1}\,(\eta_i)\right)h^p, \tag{8.18}$$

where $0<\xi_i<1$ and $0<\eta_i<1$. Next, taking the modulus of (8.18) results in

$$|\tau_n|\leq\sum_{i=0}^{k}\left(\frac{i^{p+1}}{(p+1)!}|\alpha_i|\left|y^{p+1}\,(\xi_i)\right|-\frac{i^p}{p!}|\beta_i|\left|y^{p+1}\,(\eta_i)\right|\right)h^p. \tag{8.19}$$

Now suppose that $\left|y^{(p+1)}\,(\theta)\right|\leq M_{p+1}$, then

$$|\tau_n|\leq\sum_{i=0}^{k}\left(\frac{i^{p+1}}{(p+1)!}|\alpha_i|-\frac{i^p}{p!}|\beta_i|\right)M_{p+1}h^p. \tag{8.20}$$

Therefore, by knowing k, α_i, β_i, and p and assuming that M_{p+1} exists, we can work out a bound on the truncation error. To make the linear multistep scheme formula unique, we have to normalize, which is achieved through dividing by $\sum_{i=1}^{k} \beta_i$.

Recalling the definition of the truncation error, we have

$$L(y) = \frac{dy}{dt} - f(t, y) = 0, \tag{8.21}$$

$$L_h\left(y^n\right) = \sum_{i=0}^{k} \frac{\alpha_i y^{n+i}}{h} - \sum_{i=1}^{k} \beta_i f^{n+i}. \tag{8.22}$$

We do not require $L_h(y)$ and $L(y^n)$ to be equal to zero, but we do require them to be quite small. As such, we shall use the second expression as a measure of error. Therefore, we now put the exact solution into (8.22), which results in

$$\begin{aligned} L_h(y) &= \sum_{i=0}^{k} \frac{\alpha_i y\left(t_{n+1}\right)}{h} - \sum_{i=1}^{k} \beta_i f\left(t_n, y\left(t_n\right)\right), \\ &= \sum_{i=0}^{k} \frac{\alpha_i y\left(t_n + 1\right)}{h} - \sum_{i=0}^{k} \left(\beta_i \frac{dy}{dt}\left(t_{n+1}\right)\right). \end{aligned} \tag{8.23}$$

We should note that

$$y\left(t_{n+1}\right) = y\left(t_n + ih\right), \tag{8.24}$$

$$\frac{dy}{dt}\left(t_{n+1}\right) = \frac{dy}{dt}\left(t_n + ih\right). \tag{8.25}$$

Expanding (8.24) and (8.25) via a Taylor series results in

$$y\left(t_{n+1}\right) = y\left(t_n\right) + ihy'\left(t_n\right) + \frac{(ih)^2}{2!} y''\left(t_n\right) + \cdots, \tag{8.26}$$

$$y'\left(t_{n+1}\right) = y'\left(t_n\right) + ihy''\left(t_n\right) + \frac{(ih)^2}{2!} y'''\left(t_n\right) + \cdots. \tag{8.27}$$

The truncation error is defined as

$$\varepsilon_{TE} = \frac{1}{h} C_{-1} + C_0 + hC_1 + h^2 C_2 + \cdots. \tag{8.28}$$

Therefore, collecting the terms of the same powers of h when (8.26) and (8.27) are substituted into (8.23) results in

$$C_{-1} = \sum_{i=0}^{k} \alpha_i y, \qquad = \sum_{i=0}^{k} \alpha_i y, \tag{8.29a}$$

$$C_0 = \sum_{i=0}^{k} i\alpha_i y' - \sum_{i=1}^{k} \beta_i y', \qquad = \sum_{i=0}^{k} \left(i\alpha_i - \beta_i\right) y', \tag{8.29b}$$

$$C_1 = \sum_{i=0}^{k} \frac{i^2}{2!} \alpha_i y'' - \sum_{i=1}^{k} i\beta_i y'' \qquad = \sum_{i=0}^{k} \left(\frac{i^2}{2!} \alpha_i - i\beta_i\right) y''. \tag{8.29c}$$

Given the expression in (8.29a)–(8.29c), we can state that the pth coefficient of the truncation error is

$$C_p = \sum_{i=0}^{k} \left(\frac{i^{p+1}}{(p+1)!}\alpha_i - \frac{i^p}{p!}\beta_i \right) y^{(p+1)}, \tag{8.30}$$

where the superscript $p+1$ on y refers to the $(p+1)$th derivative of y. Finally, as we let $h \to 0$ we desire the truncation error to also tend to zero; however, at the moment we have one coefficient C_{-1}, that is dependent on h^{-1} and one coefficient, C_0, that is not dependent on h, neither of which tend to zero as h to zero. This gives us two necessary conditions for the truncation error to tend to zero, which is that C_{-1} and C_0 must both be equal to zero, and thus the α_is and β_is must have the values to ensure that these two conditions are satisfied. These necessary conditions are referred to as the **consistency conditions**.

8.3 Stability

We consider the continuous case for an ordinary differential of the form

$$\begin{aligned} y' &= f(t, y), \\ y(0) &= y_0, \end{aligned}$$

where the functions on either side are nonlinear in general. The linear multistep scheme involves $y_0, y_1, y_2, y_3, \ldots$, and the truncation error is the error in the differential equation; however, we require the error in the solution.

We now consider the boundedness of the operator that *solves* the difference scheme for the following linear ordinary differential equation problem:

$$\begin{aligned} y' &= \lambda y, \\ y(0) &= y_0, \end{aligned}$$

where λ is a constant. Therefore, as shown before, the exact solution to this problem is $y = y_0 e^{\lambda t}$.

If we consider the general expression for a difference equation that solves the differential equation above with $\lambda = 0$, given by,

$$\alpha_k y^{n+k} + \alpha_{k-1} y^{n+k-1} + \cdots + \alpha_0 y^0 = 0, \tag{8.31}$$

as we wish to solve for y_n, and if we know $y^0, y^1, \ldots, y^{k-1}$, then the formula above gives the value for y^k. Therefore, if you have $y^1, y^2 \ldots, y^k$, then it is possible to obtain y^{k+1} and so on, which generates a sequence. If we take the first step, $k = 1$, then we have

$$\alpha_1 y^{n+1} + \alpha_0 y^n = 0 \Rightarrow y^{n+1} = -\frac{\alpha_0}{\alpha_1} y^n, \quad \alpha_1 \neq 0. \tag{8.32}$$

Therefore, given the first value of y, y^0, it is possible to generate y^1 and thus y^2, and as such we can define a sequence that is dependent on the previous values up to y^n as

$$y^1 = -\frac{\alpha_0}{\alpha_1} y^0,$$

$$y^2 = -\frac{\alpha_0}{\alpha_1} y^1 \equiv \left(-\frac{\alpha_0}{\alpha_1}\right)^2 y^0,$$
$$\vdots$$
$$y^n = \left(-\frac{\alpha_0}{\alpha_1}\right)^n y^0. \tag{8.33}$$

If we consider the case where $k = 2$, then we have

$$\alpha_2 y^{n+2} + \alpha_1 y^{n+1} + \alpha_0 y^n = 0. \tag{8.34}$$

Before we continue deriving properties of the difference equations, we briefly consider the following differential equation theory. Given the general form of a ordinary differential equation

$$\alpha_k \frac{d^{n+k} y}{dt^{n+k}} + \alpha_{k-1} \frac{d^{n+k-1} y}{dt^{n+k-1}} + \cdots + \alpha_0 \frac{d^n y}{dt^n} = 0, \tag{8.35}$$

then the solution to (8.35) is of the form $y = e^{mt}$. As a result of this solution, we can form the auxiliary equation as

$$\alpha_k m^{n+k} + \alpha m^{n+k-1} + \cdots + \alpha_0 m^n = 0. \tag{8.36}$$

Thus, if m is a root of (8.36), then e^{mt} is a solution. Given this information, we return to the difference equation and substitute $y^n = z^n$, which results in

$$\rho(z) = \alpha_k z^k + \alpha_{k-1} z^{k-1} + \cdots + \alpha_0 = 0. \tag{8.37}$$

Therefore, there exists up to k roots: $z_1, z_2, \ldots, z_k$, and as such $z_1^n, z_2^n, \ldots, z_k^n$ are solutions to the difference equation (8.31), which implies

$$y^n = A_1 z_1^n + A_2 z_2^n + \cdots + A_k z_k^n. \tag{8.38}$$

As an example, let us consider the difference equation

$$y^{n+2} - 3y^{n+1} + 2y^n = 0, \tag{8.39}$$

and try $y^n = z^n$, where (8.39) becomes

$$z^{n+2} - 3z^{n+1} + 2z^n = 0. \tag{8.40}$$

Factoring out z^n from (8.40) results in the following quadratic equation:

$$\rho(z) = z^2 - 3z + 2 = 0 \equiv (z-2)(z-1) \Rightarrow z_1 = 1, z_2 = 2, \tag{8.41}$$

which implies that $y^n = A_1 (1)^n + A_2 2^n = A_1 + A_2 2^n$. If we consider the differential equation from (8.1) with $\lambda = 0$, then the exact solution is $y = y_0$ and therefore $y^n = A_1 z_1^n + A_2 z_2^n + \cdots + A_k z_k^n$ is an approximate solution to y_0. However, given the values of z that we have found, we now introduce **stability** to quantify the effects of the values of z.

Definition 8.2. Stability: Suppose that $|z_i| < 1$ then $|z_i|^n$ decays when $n \to \infty$, and therefore this scheme is **stable**, if we suppose that $|z_i| = 1$ then $|z_i|^n$ is always equal to one and therefore the scheme is referred to as **neutrally stable**. Finally if $|z_i| > 1$ then $|z_i|^n \to \infty$ as $n \to \infty$ and so the associated numerical scheme is referred to as **numerically unstable**.

Therefore, given the definition given above the example scheme in (8.39) is numerically unstable.

We now explore further the stability property associated when the differential equation in (8.1) when $\lambda = 0$, which is referred to as **zero stability**. When $\lambda = 0$, then the roots of $\rho(z)$ should lie in or on the unit circle. If there are repeated roots (i.e., $z_1 = z_2$), then we replace z_2 with nz_1 such that

$$y^n = A_1 z_1^n + A_2 n z_1^n + \cdots + A_k z_k^n. \tag{8.42}$$

However, for the case where $z_1 = 1$, the approximation in (8.42) cannot decay as $n \to \infty$, and therefore the approximation will blow up. Thus for **repeated roots** we require that the roots should lie **in** the unit circle, and it is this property that is referred to as **zero stability**.

The theory that we have addressed so far has been for the scalar case, where we have a k-step recurrence relation in a scaler y, which can turn into a one-step recurrence relation in a vector, $\boldsymbol{y}$. We introduce the following definitions for $\boldsymbol{y}_n$ and $\boldsymbol{y}_{n+1}$ as

$$\boldsymbol{y}_n = \begin{pmatrix} y^n \\ y^{n+1} \\ \vdots \\ y^{n+k-1} \end{pmatrix}, \qquad \boldsymbol{y}_{n+1} = \begin{pmatrix} y^{n+1} \\ y^{n+2} \\ \vdots \\ y^{n+k} \end{pmatrix}, \tag{8.43}$$

which results in $\boldsymbol{y}_{n+1} = \mathbf{A}^n \boldsymbol{y}_n$, where

$$\mathbf{A} \equiv \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \dots & \dots & 1 \\ -\dfrac{\alpha_0}{\alpha_k} & -\dfrac{\alpha_1}{\alpha_k} & -\dfrac{\alpha_1}{\alpha_k} & \dots & \dots & \dots & -\dfrac{\alpha_{k-1}}{\alpha_k} \end{pmatrix}. \tag{8.44}$$

Therefore, we can use the fact that $\boldsymbol{y}_n = \mathbf{A}\boldsymbol{y}_{n-1}$ which results in $\boldsymbol{y}_{n+1} = \mathbf{A}\left(\mathbf{A}\boldsymbol{y}_{n-1}\right) = \mathbf{A}^2\boldsymbol{y}_{n-1}$. Repeating this expansion to $n = 0$ results in

$$\boldsymbol{y}^{n+1} = \mathbf{A}\boldsymbol{y}^0. \tag{8.45}$$

This then raises the question: Is this numerical approximation stable? We address this equation by recalling that it is possible to diagonalize a matrix to express it in terms of the matrix of eigenvectors and its inverse, along with a diagonal matrix of the eigenvalues (diagonalization) as

$$\mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}, \tag{8.46}$$

where $\mathbf{V} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k]$, and $\boldsymbol{x}_i$ are the distinct eigenvectors, and $\mathbf{D} = diag\,\{z_i\}$. Next we recall that $\mathbf{A}^n = \mathbf{AA}\ldots\mathbf{A}$, which is equivalent to

$$\mathbf{A}^n = \mathbf{VDV}^{-1}\mathbf{VDV}^{-1}\mathbf{VDV}^{-1}\ldots\mathbf{VDV}^{-1} = \mathbf{VD}^n\mathbf{V}^{-1}, \tag{8.47}$$

where $\mathbf{D}^n$ is defined by

$$\mathbf{D}^n \equiv \begin{pmatrix} z_1^n & & & \\ & z_2^n & & \\ & & \ddots & \\ & & & z_k^n \end{pmatrix}.$$

Therefore, for us to have **zero stability**, we require all of the eigenvalues z_i, $i = 1, 2, \ldots, k$ to satisfy $|z_i| \leq 1$.

Recalling the definition for determining the eigenvalues as

$$|\mathbf{A} - \lambda\mathbf{I}| = 0 \tag{8.48}$$

results in a polynomial in λ of degree k, and is equivalent to $\rho\,(\lambda)$.

If we now consider the case where we have repeated eigenvalues, the diagonal matrix becomes what is referred to as the **Jordan Normal Form**, which is defined as

$$\mathbf{J} \equiv \begin{pmatrix} z_1 & 1 & & & \\ 0 & z_1 & 0 & & \\ & & \ddots & \ddots & \\ & & & & z_k \end{pmatrix}. \tag{8.49}$$

For the repeated eigenvalues, we require $|z_i| < 1$.

8.4 Convergence

We have seen through the derivation of different properties in the previous sections of this chapter that for the general ordinary differential equation

$$\frac{dy}{dt} = f\,(t, y)\,,$$

it is possible to define a general version of the linear multistep scheme to approximate it by

$$\sum_{i=0}^{k} \frac{\alpha_i\, y_{n+i}}{h} = \sum_{i=0}^{k} \beta_i\, f_{n+i}.$$

If we first consider the $f = 0$ case, then the exact solution is $y = y_0$ everywhere, so we require $\sum_{i=0}^{k} \alpha_i\, y_{n+i} = 0$, which is the zero stability condition. Therefore, $y_n = A_1 z_1^n + A_2 z_2^n + \cdots + A_k z_k^n$, where the zs are the roots of the **first characteristic polynomial**:

$$\alpha_k z^k + \alpha_{k-1} z^{k-1} + \cdots + \alpha_0 = 0.$$

We therefore require $|z_i| \leq 1, \forall i$, else the solution blows up, therefore all roots must lie in or on the unit circle.

If we consider the case $\lambda \neq 0$, then we have the difference equation

$$\frac{1}{h}\sum_{i=0}^{k}\alpha_i y_{n+i} = \sum_{i=0}^{k}\beta_i f_{n+i} \equiv \sum_{i=0}^{k}\lambda\beta_i y_{n+i},$$
$$\Rightarrow \sum_{i=0}^{k}(\alpha_i - \lambda h\beta_i)\, y_{n+i} = 0. \tag{8.50}$$

We now have to consider whether or not $y_n = A_1 z_1^n + A_2 z_2^n + \cdots + Z_k z_k^n$ represents the exact solution, where zs are the roots of polynomial

$$a_k z^k + a_{k-1} z^{k-1} + \cdots + a_0 = 0,$$

given that the a_i are defined as $a_i = \alpha_i - \lambda h\beta_i$. When we have the situation $\lambda = 0$, then this requires $|z_i| \leq 1$, which is the zero stability condition; when we have $\lambda < 0$, then we require $|z_i| \leq 1$, which is an **absolute stability** condition; finally, when we have $\lambda > 0$, then we can allow for $|z_i| > 1$, but not by too much.

To help illustrate these properties, we consider the following example.

Example 8.3. *Consider the difference scheme*

$$\frac{y^{n+2} - y^{n+1}}{h} = \lambda\left(\frac{3}{4}y^{n+1} - \frac{1}{2}y^n\right), \tag{8.51}$$

as an approximation to the ordinary differential equation

$$\frac{dy}{dt} = \lambda y, \quad y(0) = y_0.$$

Determine whether this scheme is stable for the three different possible types of values for λ.

Solution. We start by realizing that

$$\frac{y^{n+2} - y^n}{h} = \lambda\left(\frac{3}{2}y^{n+1} - \frac{1}{2}y^n\right) \rightarrow y^n = A_1 z_1^n + A_2 z_2^n,$$

where z_1 and z_2 are the roots of

$$z^2 - z = \lambda h\left(\frac{3}{2}z - \frac{1}{2}\right) \Rightarrow z^2 - \left(1 - \frac{3}{2}\lambda h\right)z + \frac{1}{2}h = 0. \tag{8.52}$$

When we have the situation where $\lambda = 0$, then we have the zero stability case, where the exact solution is y_0. We then have $y_n = A_1 z_1^n + A_2 z_2^n$, where we have to solve the quadratic equation, $z^2 - z = 0$, for z, which has solutions $z_1 = 1$ and $z_2 = 0$, which satisfies the condition for zero stability.

Now we consider the case where $\lambda < 0$, then the solution to the differential equation is $y = y_0 e^{-\lambda t}$. As before, we form the polynomial $y_n = A_1 z_1^n + A_2 z_2^n$, where we know z_1 and z_2 are the roots of

$$z^2 - \left(1 - \frac{3}{2}\lambda h\right)z + \frac{1}{2}\lambda h,$$

and the roots of the polynomial depend continuously on the coefficients. Therefore, as λh decreases toward zero, the roots z_1 and z_2 *may* creep outside the unit circle, and as a result the approximate solution will then blow up. To avoid this happening, we have to restrict λh from going below the unit circle, which is achieved through making h **small enough**. Thus we require $|\lambda| h < 1$. If we have the case that $|\lambda|$ is big, then h may need to be very small.

Finally we consider the case where $\lambda > 0$, where the true solution of the differential equation is $y = y_0 e^{\lambda t}$. We start by forming the approximate solution in the form $y^n = A_1 z_1^n + A_2 z_2^n$; however, we have to be aware that this solution approximates exponential growth. Therefore, we require the roots to lie just outside the unit circle, but only just. For the numerical approximation to not blow up, we require z_1 and z_2 to stay within the distance h of the unit circle.

This then raises the question: What is λ? Suppose that we have two solutions

$$\frac{dy}{dt} = f(t, y), \tag{8.53}$$

$$\frac{dz}{dt} = f(t, z). \tag{8.54}$$

Combining the two equations above, we have

$$\frac{d}{dt}(y - z) = f(t, y) - f(t, z). \tag{8.55}$$

Forming the Lipschitz condition, we have

$$\frac{f(t, y) - f(t, z)}{y - z} = L, \tag{8.56}$$

where we require $|L|$ is bounded. Therefore

$$\frac{d}{dt}(y - z) = L(y - z), \tag{8.57}$$

which comes from $\frac{dy}{dt} = \lambda y$ and therefore, λ is L as far as the blow-up is concerned.

Therefore, returning to the example, we have

$$z_1 = \frac{\left(1 - \frac{3\lambda h}{2}\right) - \sqrt{\left(1 - \frac{3\lambda h}{2}\right)^2 - 2\lambda h}}{2}, \tag{8.58}$$

$$z_2 = \frac{\left(1 - \frac{3\lambda h}{2}\right) + \sqrt{\left(1 - \frac{3\lambda h}{2}\right)^2 - 2\lambda h}}{2}. \tag{8.59}$$

We can see from (8.58) that as λh tends toward zero, z_1 lies inside the unit circle which is a good property. However, from (8.59) we can see that as λh decrease, z_2 will move outside the unit circle.

8.4.1 Explicit and Implicit Numerical Scheme

We now consider two different types of schemes: **explicit** and **implicit**. The explicit scheme refers to $\beta_k = 0$, while for the implicit scheme $\beta_k \neq 0$.

The general expression for the difference equation for an **explicit** numerical scheme is given by

$$y^{n+k} = -\frac{\alpha_{k-1}}{\alpha_k} y^{n+k-1} - \frac{\alpha_{k-2}}{\alpha_k} y^{n+k-2} - \cdots + \frac{\beta_{k-1}}{\alpha_k} h f^{n+k-1}, \tag{8.60}$$

therefore, y^{n+k} is given in terms of previous values which is a **direct** relationship.

For an **implicit** numerical scheme, the associated generalized difference equation is

$$\alpha_k y^{n+k} + \alpha_{k-1} y^{n+k-1} + \cdots + \alpha_0 y^n = h\left(\beta_k f^{n+k} + \beta_{k-1} f^{n+k-1} + \cdots + \beta_0 f^n\right), \tag{8.61}$$

where we should note that $f^{n+k} \equiv f\left(t_{n+k}, y^{n+k}\right)$, which is a function of y^{n+k} and as such results in a **nonlinear equation** for y^{n+k}, implying that this method is an **indirect** relationship.

To find y^{n+k}, we would consider different iterative solvers. Two such approaches are:

- **Picard method**, which has issues with convergence and could require small values for h; and
- **Newton method**, which has issues with convergence but is not so restrictive for the values of h.

Given these different types of methods to approximate ordinary differential equations, we now consider an example of the Dahlquist convergence theorem, whose proof is beyond the scope of this book, to help illustrate the requirements for convergence of the numerical approximation.

8.4.2 Dahlquist Convergence Theorem Example

We start with the ordinary differential equation

$$y' = f(t, y), \tag{8.62}$$

and consider the difference equation approximation to (8.62) given by

$$\frac{y^{n+2} - y^{n+1}}{h} = \frac{3}{2} f^{n+1} - \frac{1}{2} f^n. \tag{8.63}$$

Substituting y in for y^n in (8.64) results in

$$\frac{y\left(t_{n+2}\right) - y\left(t_{n+1}\right)}{h} = \frac{3}{2} f\left(t_{n+1}, y\left(t_{n+1}\right)\right) - \frac{1}{2} f\left(t_n, y\left(t_n\right)\right) + \beta\tau_n. \tag{8.64}$$

Next we subtract (8.63) from (8.64), which results in

$$\begin{aligned} &\frac{y\left(t_{n+2}\right) - y^{n+2} - \left(y\left(t_{n+1}\right) - y^{n+1}\right)}{h} \\ &= \frac{3}{2}\left(f\left(t_{n+1}, y\left(t_{n+1}\right)\right) - f\left(t_{n+1}, y^{n+1}\right)\right) - \frac{1}{2}\left(f\left(t_n, y\left(t_n\right)\right) - f\left(t_n, y^n\right)\right) + \beta\tau_n. \end{aligned} \tag{8.65}$$

Applying the Lipschitz condition yields

$$L_n = \begin{cases} \dfrac{f\left(t_n, y\left(t_n\right)\right) - f\left(t_n y^n\right)}{y\left(t_n\right) - y^n} & y\left(t_n\right) \neq y^n, \\ 0 & y\left(t_n\right) = y^n. \end{cases} \tag{8.66}$$

Substituting (8.66) into the right-hand side of (8.65) results in

$$\frac{y\left(t_{n+2}\right) - y^{n+2} - \left(y\left(t_{n+1}\right) - y^{n+1}\right)}{h} = \frac{3}{2} L_{n+1}\left(y\left(t_{n+1}\right) - y^{n+1}\right) - \frac{1}{2} L_n\left(y\left(t_n\right) - y^n\right) + \beta\tau_n. \tag{8.67}$$

Now let the error be defined as $e_n = y\left(t_n\right) - y^n$, which then makes (8.67) become

$$\frac{e_{n+2} - e_{n+1}}{h} = \frac{3}{2} L_{n+1} e_{n+1} - \frac{1}{2} L_n e_n + \beta\tau_n, \tag{8.68}$$

which resembles the difference equation from (8.63). Thus the question that arises here is: How do we solve (8.68)? To answer this question, we introduce the definition for the error as

$$\boldsymbol{e}_n \equiv \begin{pmatrix} e_n \\ e_{n+1} \end{pmatrix}, \qquad \boldsymbol{e}_{n+1} \equiv \begin{pmatrix} e_{n+1} \\ e_{n+2} \end{pmatrix},$$

which enables (8.68) to be rewritten as

$$\boldsymbol{e}_{n+1} = \left(\mathbf{A} + h\mathbf{B}_n\right)\boldsymbol{e}_n + h\beta\boldsymbol{\tau}_n, \tag{8.69}$$

where

$$\mathbf{A} \equiv \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{B}_n \equiv \begin{pmatrix} 0 & 0 \\ -\dfrac{L_n}{2} & \dfrac{3L_{n+1}}{2} \end{pmatrix}, \quad \boldsymbol{\tau}_n \equiv \begin{pmatrix} 0 \\ \tau_n \end{pmatrix}.$$

Now, forming the sequence of steps starting at $t = 0$, we have

$$\begin{aligned} \boldsymbol{e}_1 &= \left(\mathbf{A} + h\mathbf{B}_0\right)\boldsymbol{e}_0 + h\beta\boldsymbol{\tau}_0, \\ \boldsymbol{e}_2 &= \left(\mathbf{A} + h\mathbf{B}_1\right)\boldsymbol{e}_1 + h\beta\boldsymbol{\tau}_1, \\ &\vdots \\ \boldsymbol{e}_n &= \left(\mathbf{A} + h\mathbf{B}_{n-1}\right)\boldsymbol{e}_{n-1} + h\beta\boldsymbol{\tau}_{n-1}. \end{aligned} \tag{8.70}$$

The next step is to multiply the ith equation by $\prod_{i=1}^{n-1}\left(\mathbf{A} + h\mathbf{B}_i\right)$, which comes about through substituting $\boldsymbol{e}_1$ into the expression for $\boldsymbol{e}_2$ above, which results in

$$\boldsymbol{e}_n = \prod_{i=0}^{n-1}\left(\mathbf{A} + h\mathbf{B}_i\right)\boldsymbol{e}_0 + \beta h \sum_{i=1}^{n}\prod_{i=1}^{n-1}\left(\mathbf{A} + h\mathbf{B}_i\right)\boldsymbol{\tau}_{i-1}. \tag{8.71}$$

Because we know the exact initial conditions, and the scheme matches them at the initial time, implies that the initial error is $\boldsymbol{e}_0 = \mathbf{0}$. Thus we have

$$\boldsymbol{e}_n = \beta h \sum_{i=1}^{n}\prod_{i=1}^{n-1}\left(\mathbf{A} + h\mathbf{B}_i\right)\boldsymbol{\tau}_{i-1}. \tag{8.72}$$

Applying a general norm to (8.72) results in

$$\|\boldsymbol{e}_n\| \leq \sum_{i=1}^{n} \left\| \prod_{i=1}^{n-1} (\mathbf{A} + h\mathbf{B}_i) \right\| \|\beta h\| \|\boldsymbol{\tau}_{i-1}\|, \tag{8.73}$$

where it is possible to simplify (8.73) to

$$\|\boldsymbol{e}_n\| \leq \beta K t_n e^{MKt_n} \tau, \tag{8.74}$$

where $\boldsymbol{e} = y(t_n) - y^n$ is referred to as the **global error** and as such the inequality in (8.74) is referred to as the **global error bound**, where $t_n = nh$, $\beta = \sum_{i=1}^{k} \beta i$, τ is the bound on the truncation error. However, what are K and M? These are defined as

$$K = \max\left\{1, \|\mathbf{A}^n\|\right\}, \qquad M = \max\left\{\|\mathbf{B}_n\|\right\}, \tag{8.75}$$

where in general $\mathbf{A}$ and $\mathbf{B}_n$ are given by

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \vdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \\ -\dfrac{\alpha_0}{\alpha_k} & -\dfrac{\alpha_1}{\alpha_k} & -\dfrac{\alpha_2}{\alpha_k} & \cdots & -\dfrac{\alpha_{k-1}}{\alpha_k} \end{pmatrix},$$

$$\mathbf{B}_n = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \dfrac{\beta_0}{\alpha_k} L_n & \dfrac{\beta_1}{\alpha_k} L_{n+1} & \dfrac{\beta_2}{\alpha_k} L_{n+2} & \cdots & \dfrac{\beta_{k-1}}{\alpha_k} L_{n+k-1} \end{pmatrix}.$$

Returning to our example, the difference equation is given by

$$\frac{y^{n+2} - y^{n+1}}{h} = \frac{3}{2} f^{n+1} - \frac{1}{2} f^n,$$

as such the associated αs and βs for this example are

$$\begin{aligned} \alpha_2 &= 1, & \alpha_1 &= -1, & \alpha_0 &= 0, \\ \beta_2 &= 0, & \beta_1 &= \tfrac{3}{2}, & \beta_0 &= -\tfrac{1}{2}, \end{aligned}$$

therefore $\beta = \sum_{i=0}^{2} \beta_i = 1$,

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \quad \mathbf{B}_n = \begin{pmatrix} 0 & 0 \\ -\frac{1}{2} L_n & \frac{3}{2} L_{n+1} \end{pmatrix}.$$

The next coefficient of the bound that we consider is M:

$$M = \max_n \left\| \begin{pmatrix} 0 & 0 \\ -\frac{1}{2}L_n & \frac{3}{2}L_{n+1} \end{pmatrix} \right\| \equiv \left\| \begin{pmatrix} 0 & 0 \\ -\frac{1}{2}L_n & \frac{3}{2}L_{n+1} \end{pmatrix} \right\|_\infty,$$
$$= \frac{1}{2}|L_n| + \frac{3}{2}|L_{n+1}| \leq \frac{1}{2}L + \frac{3}{2}L = 2L.$$

Now we consider the term K:

$$K = \max_n \left\{ 1, \left\| \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}^n \right\| \right\}.$$

We recall that we can diagonalize $\mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$, such that $\mathbf{A}^n = \mathbf{V}\mathbf{D}^n\mathbf{V}^{-1}$. We require that the two eigenvalues satisfy the conditions $|\lambda_1| \leq 1$ and $|\lambda_2| \leq 2$,

$$\|\mathbf{A}^n\| \leq \|\mathbf{V}\|\|\mathbf{D}^n\|\|\mathbf{V}^{-1}\| \leq \|\mathbf{V}\|\|\mathbf{V}^{-1}\|,$$

where we have used the fact that $\|\mathbf{D}^n\| < 1$. Solving for the eigenvalues of $\mathbf{A}$, we have

$$det\,|\mathbf{A} - \lambda\mathbf{I}| = 0 \Rightarrow \begin{vmatrix} -\lambda & 1 \\ 0 & 1-\lambda \end{vmatrix} = 0 \Rightarrow \lambda(1-\lambda) = 0 \Rightarrow \lambda_1 = 1, \lambda_2 = 0.$$

Next, solving for the eigenvectors of $\mathbf{A}$, we have

$$\boldsymbol{e}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \boldsymbol{e}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \Rightarrow \mathbf{V} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \Rightarrow \|\mathbf{V}\| = 2,$$
$$\mathbf{V}^{-1} = \begin{pmatrix} 0 & 1 \\ 1 & -1 \end{pmatrix} \Rightarrow \|\mathbf{V}\| = 2,$$

therefore $\|\mathbf{A}^n\| \leq 4 \Rightarrow K = 4$.

Finally, collecting all of the coefficients derived above, we have a bound of the global error for this scheme as

$$\|\boldsymbol{e}_n\| \leq 1 \times 4te^{8Lt\tau}. \tag{8.76}$$

The error bound in (8.76) is a very pessimistic and crude approximation to the bound on the error, but we now know that the error will not be larger than the constraint in (8.76).

Therefore, in summary for the linear multistep method, we have that:

- truncation error—consistency, order of accuracy;
- stability—roots inside or on the unit circles; and
- convergence—global error bounded and tends to zero in the limit, that is, $e_n \to 0$ because $\tau \to 0$ and $h \to 0$.

We now present some specific one-step methods to be aware of that are used for different modeling:

- **Forward Euler**: $\dfrac{y^{n+1} - y^n}{h} = f^n$ or can be written as $y^{n+1} = y^n + hf^n$.
- **Backward Euler**: $\dfrac{y^{n+1} - y^n}{h} = f^{n+1}$.

- **Trapezoidal**: $\dfrac{y^{n+1}-y^n}{h}=\dfrac{1}{2}\left(f^n+f^{n+1}\right)$.

Exercise 8.4. *Verify that the numerical scheme given by*

$$y_{n+2}-\frac{3}{2}y_{n+1}+\frac{1}{2}y_n=h\left(\frac{5}{4}f_{n+1}-\frac{3}{4}f_n\right),$$

is an approximation to the differential equations $\dfrac{dy}{dt}=f(t,y)$*, is consistent and zero-stable. Find a bound on the truncation error* τ_n*, assuming that* y *has a bounded third derivative.*

Construct matrices $\mathbf{A}$ *and* $\mathbf{B}_n$ *such that* $\boldsymbol{e}_{n+1}=(\mathbf{A}+h\mathbf{B})\,\boldsymbol{e}_n+\mathbf{b}_n$*, where* $\boldsymbol{e}_n^T=\begin{pmatrix} e_n & e_{n+1}\end{pmatrix}$, $\mathbf{b}^T=\begin{pmatrix}0 & h\tau_n\end{pmatrix}$*, and the error is defined as* $e_n=y(t_n)-y_n$.

Next find the matrix $\mathbf{V}$ *such that the product,* $\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$*, is a diagonal matrix and hence find a bound on* $\|\mathbf{A}^n\|_\infty$*. Hence find a bound on the error* $\|\boldsymbol{e}_n\|_\infty$ *at* $t=nh$*, using the convergence theorem just mentioned.*

8.5 Runge-Kutta Schemes

We move on to consider the second set of approximation methods mentioned earlier; the Runge-Kutta methods. These methods can be either explicit or implicit. We start with the family of explicit Runge-Kutta methods.

8.5.1 Explicit Runge-Kutta Methods

The Runge-Kutta methods are based on forming a one-step nonlinear approximation to the differential equation, instead of a linear method which the multistep methods are. To obtain this nonlinear approximation, we start by approximating y and y^{n+1} via Taylor series expansions as

$$y(t_n+h)=y(t_n)+hy'(t_n)+\frac{h^2}{2!}y''(t)+\cdots, \tag{8.77}$$

$$y^{n+1}=y^n+hf^n+\frac{h^2}{2!}\left(\frac{\partial f}{\partial t}+f\frac{\partial f}{\partial y}\right)+\frac{h^3}{3!}\cdots, \tag{8.78}$$

where $y'=f(t,y)$ and $y''=\frac{\partial f}{\partial t}+\frac{\partial f}{\partial y}\frac{dy}{dt}$.

Therefore, we have

$$y^{n+1}=y^n+h\tilde{f}^n, \tag{8.79}$$

where $\tilde{f}^n$ is an estimate of the right-hand side of the differential equation. The general form of a Runge-Kutta approximation is given by

$$y^{n+1}=y^n+h\sum_{i=1}^{s}b_ik_i, \tag{8.80}$$

where $k_i=f\left(t_n+c_ih,\,y+a_{ij}k_{i-1}\right)$, which expands as

$$
\begin{aligned}
k_1 &= f\left(t_n, y^n\right), \\
k_2 &= f\left(t_n + c_2 h, y^n + h\left(a_{21}, k_1\right)\right), \\
k_3 &= f\left(t_n + c_3 h, y^n + h\left(a_{31} k_1 + a_{32} k_2\right)\right), \\
&\vdots \\
k_s &= f\left(t_n + c_s h, y^n + h\left(a_{s1} k_1 + a_{s2} k_2 + \cdots + a_{ss-1} k_{s-1}\right)\right).
\end{aligned} \tag{8.81}
$$

We can see from (8.81) that the Runge-Kutta schemes are based upon linear combinations of the slopes of the function. The matrix $\left[a_{ij}\right]$ is referred to as the **Runge-Kutta matrix**, while the b_is are referred to as the **weights** and the c_is are referred to as the **nodes**. There is a popular tabulation approach to visualize the coefficients, referred to as the **Butcher tableau**, where for (8.81) this is

$$
\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_s & a_{s1} & a_{s2} & \cdots & a_{ss-1} & \\
\hline
 & b_1 & b_2 & \cdots & b_{s-1} & b_s
\end{array} \tag{8.82}
$$

which can also be displayed as $\begin{array}{c|c} \boldsymbol{c} & \mathbf{A} \\ \hline & \boldsymbol{b}^T \end{array}$.

The Runge-Kutta general scheme presented earlier is referred to as the *explicit* Runge-Kutta methods, as the k_nth term, for $n = 1, 2, \ldots, s$, is dependent on all of the previous k_is terms for $i = 1, 2, \ldots, n-1$. The explicit Runge-Kutta schemes, as we can see from (8.82), implies that $\mathbf{A}$ is a strictly lower triangular array, which means that all the non-zero values are below the diagonal entries. There is a class of Runge-Kutta methods that are called **semi-implicit** Runge-Kutta scheme in that the array $\mathbf{A}$ is a lower triangular matrix (i.e., there are values on the diagonal but not in the upper triangular part of the matrix).

We have already presented the simplest Runge-Kutta method, which corresponds to a one-stage Runge-Kutta method, and that is the forward Euler scheme. A commonly used second-order method is the **midpoint method**, which is given by

$$
y^{n+1} = y^n + hf\left(t_n + \frac{1}{2}h, y^n + \frac{1}{2}hf\left(t_n, y^n\right)\right), \tag{8.83}
$$

where the associated Butcher tableau is

$$
\begin{array}{c|cc}
0 & & \\
\frac{1}{2} & \frac{1}{2} & \\
\hline
 & 0 & 1
\end{array}
$$

8.5.2 Consistency and Stability of Explicit Runge-Kutta Methods

As we saw with the linear multistep scheme, the requirement for a scheme to be consistent is that the truncation error tends to zero as $h \to 0$. Therefore, the truncation error for the general form of a

Runge-Kutta scheme is

$$\tau_n \equiv \frac{y\left(t_{n+1}\right) - y\left(t_n\right)}{h} - \sum_{i=1}^{s} b_i f\left(t_n + c_i h, y^n + h\sum_{j=1}^{s} a_{ij} k_j\right),$$
$$= \left(1 - \sum_{i=1}^{s} b_i\right) y'_n + \frac{h}{2} y'' - \sum_{i=1}^{s} b_i \left(c_i h f_t + h \sum_{j=1}^{s} a_{ij} f f_y\right) + O\left(h^2\right), \tag{8.84}$$

where $f_t \equiv \frac{\partial f}{\partial t}$ and $f_y \equiv \frac{\partial f}{\partial y}$. Therefore, the first-order condition for the general Runge-Kutta scheme to be consistent is $\sum_{i=1}^{s} b_i = 1$. This removes the term in (8.84) that is not dependent on h. To make the Runge-Kutta scheme consistent with the second-order derivative term $y'' = f_t + f f_y$, we require $\frac{1}{2} = \sum_{i=1}^{s} b_i c_i = \sum_{i=1}^{s} b_i \sum_{j=1}^{s} a_{ij}$.

To help illustrate these consistency requirements, let us consider the explicit case where $s = 2$; here there are two possible Runge-Kutta formulations that satisfy the consistency conditions $b_1 + b_2 = 1$ and $\frac{1}{2} = b_1 c_1 + b_2 c_2 = b_2 a_{21}$;

$$\begin{array}{c|cc} 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ \hline & 0 & 1 \end{array} \Rightarrow y^{n+1} = y_n + hf\left(t_n + \frac{h}{2}, y^n + \frac{h}{2} f^n\right),$$

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array} \Rightarrow y^{n+1} = y^n + \frac{h}{2}\left(f^n + f\left(t_n, y^n + hf^n\right)\right).$$

The first scheme is the forward Euler method, where the second scheme is the improved Euler method.

We can clearly see that the two Runge-Kutta schemes above are consistent with the second-order term in the Taylor series expansion of the differential equation.

If we now consider a Taylor series expansion truncated at the third-order term, then we have

$$y^{n+1} \approx y^n + hy'_n + \frac{h^2}{2!} y''_n + \frac{h^3}{3!} y'''_n. \tag{8.85}$$

For stability we substitute the solution, $y' = \lambda y$ into (8.85) and obtain

$$y^{n+1} \approx \left(1 + h\lambda + \frac{h^2\lambda^2}{2!} + \frac{h^3\lambda^3}{3!}\right) y_n, \tag{8.86}$$

where the stability depends on the amplification factors, the term multiplying y_n in (8.86) lying in, or on, the unit circle. Therefore, we now have what is referred to as the **stability polynomial**, where for third-order schemes this is

$$\rho(z) = z - \left(1 + h\lambda + \frac{h^2\lambda^2}{2!} + \frac{h^3\lambda^3}{3!}\right). \tag{8.87}$$

As such the stability condition for the third-order schemes is given by

$$\left|1 + h\lambda + \frac{h^2\lambda^2}{2!} + \frac{h^3\lambda^3}{3!}\right| \leq 1. \tag{8.88}$$

In Fig. 8.2 we have plotted the stability regions for the Runge-Kutta schemes up to the fourth-order scheme. As we can see from Fig. 8.2, when $s = 1$, the stability region is in the negative part of complex plane and is of the form of an ellipse. Considering the second-order methods, the stability region has stretched and includes a larger range of values. For the third-order Runge-Kutta methods we have a larger region than for the two lower-order schemes, but now there is some creepage into the positive values for the real number component. Finally, when we consider the stability region for the fourth-order methods, we see that it is possible for a larger group of the complex components of the roots of (8.88) to have a positive component.

8.5.3 Derivation of the Fourth-Order Runge-Kutta Scheme

The fourth-order Runge-Kutta scheme is quite often used in toy problems in atmospheric data assimilation; one of the reasons for this is that it is relative easily to code, but guarantees fourth-order convergence, and as we can see from the stability region plots in Fig. 8.2, the fourth-order

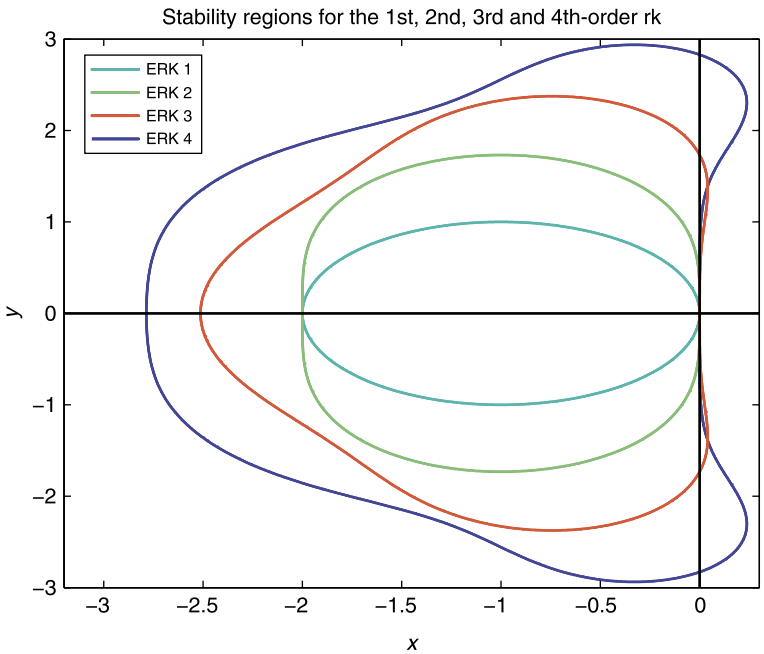


FIGURE 8.2

Plot of the stability polynomial for the first-, second-, third-, and fourth-order schemes.

Runge-Kutta scheme has a larger range of possible step sizes that are guaranteed to keep the scheme stable.

When we considered the second-order Runge-Kutta schemes earlier we showed that the scheme was consistent with the different order of derivatives in the Taylor series expansion of $y(t+h)$. Thus, the starting point for the derivation of the fourth-order Runge-Kutta scheme is to evaluate the general formula

$$y^{n+1} = y^n + h\sum_{i=1}^{s} b_i k_i,$$
$$k_i = f\left(t_n + c_i h, y^n + h\sum_{j=1}^{s} a_{ij}k_j\right),$$

where the fourth-order scheme is going to be evaluated at the starting point, the midpoint, and the end point of any time interval $[t_n, t_n + h]$. This implies using the following values for the coefficients of a_{ij} and c_i as

$$c_1 = 0 \quad c_2 = \tfrac{1}{2} \quad c_3 = \tfrac{1}{2} \quad c_4 = 1,$$
$$a_{21} = \tfrac{1}{2} \quad a_{32} = \tfrac{1}{2} \quad a_{43} = 1,$$

where the $a_{ij} = 0$ for the other values of i and j. To make things easier with the derivation, as you can see from the $k_i = f\left(t_n + c_i h, y^n + h\sum_{j=1}^{s} a_{ij}k_j\right)$ term, there are going to be up to four terms in each k_i, we introduce the following notation:

$$y_1^{n+1} \equiv y^n + hf\left(t_n, y^n\right),$$
$$y_1^{n+\frac{1}{2}} \equiv \frac{y^n + y_1^{n+1}}{2},$$
$$y_2^{n+1} \equiv y^n + hf\left(t_n + \frac{h}{2}, y_1^{n+\frac{1}{2}}\right),$$
$$y_2^{n+\frac{1}{2}} \equiv \frac{y^n + y_2^{n+1}}{2},$$
$$y_3^{n+1} \equiv y^n + hf\left(t_n + \frac{h}{2}, y_2^{n+\frac{1}{2}}\right),$$

which leads to the following expressions for the k_is:

$$k_1 \equiv f\left(t_n, y^n\right),$$
$$k_2 \equiv f\left(t_n + \frac{h}{2}, y_1^{n+\frac{1}{2}}\right),$$
$$k_3 \equiv f\left(t_n + \frac{h}{2}, y_2^{n+\frac{1}{2}}\right),$$
$$k_4 \equiv f\left(t_n + h, y_3^{n+1}\right).$$

Next we have to derive formulas for the k_is in terms of differential operators, which are given by,

$$\begin{aligned}
k_2 &= f\left(t_n + \frac{h}{2}, y_1^{n+\frac{1}{2}}\right) = f\left(t + \frac{h}{2}, y^n + \frac{h}{2}k_1\right), \\
&= f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt} f\left(t_n, y^n\right), \qquad (8.89) \\
k_3 &= f\left(t_n + \frac{h}{2}, y_2^{n+\frac{1}{2}}\right) = f\left(t_n + \frac{h}{2}, y^n + \frac{h}{2} f\left(t_n + \frac{h}{2}, y^n + \frac{h}{2}k_1\right)\right), \\
&= f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt}\left(f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt} f\left(t_n, y^n\right)\right), \\
&= f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt}\left(f\left(t_n, y^n\right)\right) + \frac{h^2}{4}\frac{d^2}{dt^2}\left(f\left(t_n, y^n\right)\right), \qquad (8.90) \\
k_4 &= f\left(t_n + h, y_3^{n+1}\right) = f\left(t_n + h, y^n + hf\left(t_n + \frac{h}{2}, y^n + \frac{k_2}{2}\right)\right), \\
&= f\left(t_n + h, y^n + hf\left(t_n + \frac{h}{2}, y^n + \frac{h}{2} f\left(t_n + \frac{h}{2}, y^n + \frac{h}{2} f\left(t_n, y^n\right)\right)\right)\right), \\
&= f\left(t_n, y^n\right) + h\frac{d}{dt}\left(f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt}\left(f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt} f\left(t_n, y^n\right)\right)\right), \\
&= f\left(t_n, y^n\right) + h\frac{d}{dt} f\left(t_n, y^n\right) + \frac{h^2}{2}\frac{d^2}{dt^2} f\left(t_n, y^n\right) + \frac{h^3}{4}\frac{d^3}{dt^3} f\left(t_n, y^n\right). \qquad (8.91)
\end{aligned}$$

Now we substitute (8.89)–(8.91) into the general expression for the Runge-Kutta methods to obtain the b_is values to complete the expression for the fourth-order Runge-Kutta scheme; thus

$$\begin{aligned}
y_{n+1} = y^n &+ h\left(b_1 f\left(t_n, y^n\right) + b_2\left(f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt} f\left(t_n, y^n\right)\right)\right. \\
&+ b_3\left(f\left(t_n, y^n\right) + \frac{h}{2}\frac{d}{dt} f\left(t_n, y^n\right) + \frac{h^2}{4}\frac{d^2}{dt^2} f\left(t_n, y^n\right)\right) \\
&\left.+ b_4\left(f\left(t_n, y^n\right) + h\frac{d}{dt} f\left(t_n, y^n\right) + \frac{h^2}{2}\frac{d^2}{dt^2} f\left(t_n, y^n\right) + \frac{h^3}{4}\frac{d^3}{dt^3} f\left(t_n, y^n\right)\right)\right) \\
&+ \mathcal{O}\left(h^5\right). \qquad (8.92)
\end{aligned}$$

Collecting all similar terms of the powers of h, and powers of the derivatives of f, results in

$$\begin{array}{c|l}
hf\left(t_n, y^n\right) & b_1 + b_2 + b_3 + b_4 \\
h^2\frac{d}{dt} f\left(t_n, y^n\right) & \frac{b_2}{2} + \frac{b_3}{2} + b_4 \\
h^3\frac{d^2}{dt^2} f\left(t_n, y^n\right) & \frac{b_3}{4} + \frac{b_4}{2} \\
h^4\frac{d^3}{dt^3} f\left(t_n, y^n\right) & \frac{b_4}{4}.
\end{array} \qquad (8.93)$$

To ascertain what the values of the b_is should be, we consider the Taylor series expansion of $y(t_n + h)$ to fourth order of h about $y(t_n)$, using the relationship that $\frac{d}{dt}y(t_n) \equiv f(t_n, y^n)$, which yields

$$\begin{aligned} y(t_n + h) &= y(t_n) + h\frac{d}{dt}y(t_n) + \frac{h^2}{2}\frac{d^2}{dt^2}y(t_n) + \frac{h^3}{6}\frac{d^3}{dt^3}y(t_n) + \frac{h^4}{24}\frac{d^4}{dt^4}y(t_n) + \mathcal{O}\left(h^5\right), \\ &\equiv y(t_n) + hf\left(t_n, y^n\right) + \frac{h^2}{2}\frac{d}{dt}f\left(t_n, y^n\right) + \frac{h^3}{6}\frac{d^2}{dt^2}f\left(t_n, y^n\right) + \frac{h^4}{24}\frac{d^3}{dt^3}f\left(t_n, y^n\right) + \mathcal{O}\left(h^5\right). \end{aligned} \quad (8.94)$$

Therefore, equating (8.92) with (8.94) enables us to obtain the following four simultaneous equations for four unknowns:

$$\begin{aligned} b_1 + b_2 + b_3 + b_4 &= 1, \\ \frac{b_2}{2} + \frac{b_3}{2} + b_4 &= \frac{1}{2}, \\ \frac{b_3}{4} + \frac{b_4}{2} &= \frac{1}{6}, \\ \frac{b_4}{4} &= \frac{1}{24}. \end{aligned}$$

Thus we have $b_1 = \frac{1}{6}$, $b_2 = \frac{1}{3}$, $b_3 = \frac{1}{3}$, and $b_4 = \frac{1}{6}$, and the final Butcher tableau is

$$\begin{array}{c|cccc} 0 & & & & \\ \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array}$$

Exercise 8.5. *Find the conditions that make the three-stage explicit Runge-Kutta scheme*

$$y_{n+1} = y_n + \sum_{i=1}^{3} b_i k_i,$$

with

$$\begin{aligned} k_1 &= f(t, y), \\ k_2 &= f(t + hc_2, y + hc_2 k_1), \\ K_3 &= f(t + hc_3, y + h(c_3 - a_{32})k_1 + ha_{32}k_2), \end{aligned}$$

second order. Derive the associated stability polynomial for this scheme.

8.6 Numerical Solutions to Initial Value Partial Differential Equations

A general partial differential equation can be expressed using many different forms of differential operators. As such, the solution we are seeking, $u = u(t, x, y, \ldots)$, and the differential operators acting

on u are given by

$$u_t \equiv \frac{\partial u}{\partial t}, \quad u_x \equiv \frac{\partial u}{\partial x}, \quad u_{xx} \equiv \frac{\partial^2 u}{\partial x^2}, \tag{8.95}$$

which leads to a general expression for the initial value partial differential equation as

$$u_t = F\left(t, x, y, \ldots, u_x, u_y \ldots, u_{xx}, u_{yy}, u_{xy}, \ldots\right). \tag{8.96}$$

If we consider just the one space dimension here, then (8.96) simplifies to

$$u_t = F\left(t, x, u_x, u_{xx}, \ldots\right). \tag{8.97}$$

An example where the partial differential equation is a function of u_x would be an advection, convection, or first-order wave, equation, given by

$$u_t = -cu_x. \tag{8.98}$$

It is also possible that the partial differential equation is only a function of u_{xx}; for example the diffusion, or heat equation, given by

$$u_t = \sigma u_{xx}. \tag{8.99}$$

As mentioned before, there are different types of partial differential equations but in this section we shall only consider parabolic partial differential equations (e.g., $u_t = u_{xx}$) and hyperbolic partial differential equations (e.g., $u_{tt} = u_{xx}$ and $u_t = u_x$). Therefore, for partial differential equations we shall be discretizing in both space and time. As we are dealing with initial value problems in this chapter, we have initial conditions, $u\left(0, x, y, \ldots\right) = f\left(x, y, \ldots\right)$, for (8.96). Therefore, the function f has to be also discretized, with some loss of information.

8.6.1 Heat Equation

The first partial differential equation we consider is the heat equation, where in Fig. 8.3 we have provided a schematic of the flow of heat. The heat balance gives $u_t = \frac{\partial}{\partial x}\left(\sigma \frac{\partial u}{\partial x}\right)$, which simplifies to (8.99) if σ is a constant and is therefore not a function of x. The question is then: does an analytical solution

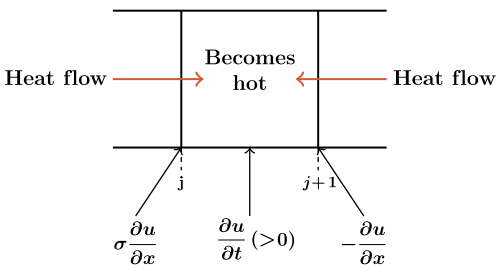


FIGURE 8.3

Schematic of the heat balance equation derivation.

to the heat balance equations exist? If it possible to apply Fourier methods, separation of variables or Green's functions to solve, where $u = e^{-k^2 t}(\cos kt + \sin kt)$. In this formulation we have arrived at $u_t = \sigma u_{xx}$ from the full variable differential equation

$$a\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(K\frac{\partial u}{\partial x}\right), \tag{8.100}$$

where u is temperature, a is the heat capacity, and K is the thermal conductivity; therefore, we assume that a and K are constant such that this then makes $\sigma = \frac{K}{a}$. At the initial time we have the initial condition $u(0, x) = f(x)$ which is a function of x. Unlike with the ordinary differential equations, we now have boundary conditions in the x-directions to constrain the solution. For example, we have the boundary conditions, $u(t, 0) = u(t, \pi) = 0$. Particular analytical solutions do not satisfy the boundary conditions. However, there does exist infinite series solutions, but these approaches have slow convergence, and can cause problems.

8.6.2 Numerical Approach

For the ordinary differential equation approaches we were considering graphs where we were evaluating the function y at different points along the time axis. For partial differential equations we have a grid (see Fig. 8.4), where the n indices is associated with time and the j indices are associated with the spatial coordinates. For partial differential equations we denote the numerical solution at the n time step and jth space location as u_j^n, and the change in the time direction between two time steps is denoted, Δt, and between to spatial points by Δx.

We now introduce numerical approximations to the different differential operators. For the spatial derivative we have the following approximation valid at the halfway between two adjacent x points, as

$$\left(\frac{\partial u}{\partial x}\right)_{j+\frac{1}{2}} \approx \frac{u_{j+1}^n - u_j^n}{\Delta x}. \tag{8.101}$$

For the first-order time derivative we have

$$\frac{\partial u}{\partial t} \approx \frac{u_j^{n+1} - u_j^n}{\Delta t}. \tag{8.102}$$

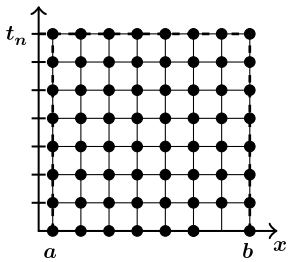


FIGURE 8.4

Visualization of the plane view of the numerical grid for partial differential equations.

For the second-order spatial derivative we consider the difference between two numerical approximations to the first-order derivative valid at the points $j+\frac{1}{2}$ and $j-\frac{1}{2}$, which makes this approximation to the second-order derivative valid at the jth location. Therefore, we have

$$\begin{aligned}\frac{\partial^2 u}{\partial x^2} &\approx \frac{\left(\frac{\partial u}{\partial x}\right)_{j+\frac{1}{2}} - \left(\frac{\partial u}{\partial x}\right)_{j-\frac{1}{2}}}{\Delta x} \approx \frac{\frac{u^n_{j+1}-u^n_j}{\Delta x} - \frac{u^n_j - u^n_{j-1}}{\Delta x}}{\Delta x}, \\ &\approx \frac{u^n_{j+1} - 2u^n_j - u^n_{j-1}}{\Delta x^2}. \end{aligned} \tag{8.103}$$

Thus, given (8.102) and (8.103), we can now state an explicit numerical scheme approximation to the heat equation as

$$\frac{u^{n+1}_j - u^n_j}{\Delta t} = \frac{\sigma}{\Delta x^2}\left(u^n_{j+1} - 2u^n_j + u^n_{j-1}\right). \tag{8.104}$$

If we have initial conditions, $u(t_0)$, then we know u^0_j. Given u^0_j, it is then possible to work out $n=1$. If we keep repeating the steps following (8.104), then we have a set of rules to obtain u^n_j given u^0_j.

We now address the area of accuracy, which is again obtained through considering the truncation error, and stability, through considering the amplification factor. We consider the accuracy first. The starting point is to consider the two operators, $L(u)$ that represents the continuous partial differential equations, and $L_h(u)$ that represents the numerical approximation. For the heat equation, these two operators are

$$L(u) = u_t - \sigma u_{xx} = 0, \tag{8.105a}$$

$$L_h\left(u^n_j\right) = \frac{u^{n+1}_j - u^n_j}{\Delta t} - \frac{\sigma}{\Delta x^2}\left(u^n_{j+1} - 2u^n_j + u^n_{j-1}\right) = 0. \tag{8.105b}$$

Given the definitions in (8.105a) and (8.105b) we would expect $L\left(u^n_j\right) \neq 0$ and $L_h(u) \neq 0$, but are expected to be small.

Following on from definition of the truncation error for the ordinary differential equations, we can define the truncation error for the partial differential equations as

$$\tau^n_j = L_h(u) \equiv \frac{u\left(t_{n+1}, x_j\right) - u\left(t_n, x_j\right)}{\Delta t} - \frac{\sigma}{\Delta x^2}\left(u\left(t_n, x_j+1\right) - 2u\left(t_n, x_j\right) + u\left(t_n, x_{j-1}\right)\right). \tag{8.106}$$

The next step is to apply a Taylor series expansion of u about $\left(x_j, t^n\right)$, so that $t_{n+1} = t_n + \Delta t$, and $x_{j\pm1} = x_j \pm \Delta x$, which results in the following expression with respect to the time component

$$\begin{aligned} u\left(t_{n+1}, x_j\right) &= u\left(t_n + \Delta t, x_j\right) = u\left(t_n, x_j\right) + \Delta t u_t\left(t_n, x_j\right) + \cdots \\ &= u + \Delta u_t + \frac{(\Delta t)^2}{2!}u_{tt} + \cdots, \end{aligned} \tag{8.107}$$

where for the spatial components we have

$$u\left(t_n, x_{j\pm1}\right) = u\left(t_n, x \pm \Delta x\right) = u\left(t_n, x_j\right) \pm \Delta x u_x\left(t_n, x_j\right) + \frac{(\Delta x)^2}{2!}u_{xx}\left(t_n, x_j\right)$$

$$\pm \frac{(\Delta x)^3}{3!} u_{xxx}(t_n, x_j) + \frac{(\Delta x)^4}{4!} u_{xxxx}(t_n, x_j) \pm \cdots,$$
$$= u \pm \Delta x u_x + \frac{(\Delta x)^2}{2!} \pm \frac{(\Delta x)^3}{3!} u_{xxx} + \frac{(\Delta x)^4}{4!} \pm \cdots. \tag{8.108}$$

Now, substituting (8.107) and (8.108) into (8.106) results in

$$\begin{aligned}\tau_j^n = & \frac{u + \Delta t u_t + \frac{(\Delta t)^2}{2!} u_{tt} + \cdots - u}{\Delta t} \\ & - \frac{\sigma}{(\Delta x)^2} \left(u + \Delta u_x + \frac{(\Delta x)^2}{2!} u_{xx} + \frac{(\Delta x)^3}{3!} u_{xxx} + \frac{(\Delta x)^4}{4!} u_{xxxx} - 2u \right. \\ & \left. + u - \Delta x u_x + \frac{(\Delta x)^2}{2!} u_{xx} - \frac{(\Delta x)^3}{3!} u_{xxx} + \frac{(\Delta x)^4}{4!} + \cdots \right), \\ = & u_t - \sigma u_{xx} + \frac{\Delta t}{2!} u_{tt} - \sigma \frac{(\Delta x)^2}{12} u_{xxxx} + \cdots. \end{aligned} \tag{8.109}$$

The first term in (8.109) is equal to zero as this is the original partial differential equation, therefore the truncation error for this explicit scheme is

$$\tau_j^n = \frac{\Delta t}{2} u_{tt} - \sigma \frac{(\Delta x)^2}{12} u_{xxxx} + \cdots. \tag{8.110}$$

Therefore the truncation error $\tau_j^n \to 0$ as $\Delta t \to 0$ and $\Delta x \to 0$, and thus the explicit scheme is consistent, and is first order in time and second order in space.

Two other possible consistent schemes to the heat equations are an implicit scheme, and the Crank-Nicolson scheme, which defined by:

- *Implicit scheme*:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2} \delta^2 u_j^{n+1}, \qquad \tau_j^n = \mathcal{O}(\Delta t) + \mathcal{O}\left((\Delta x)^2\right); \tag{8.111}$$

- *Crank-Nicolson*:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2} \left[\frac{1}{2} \delta^2 u_j^{n+1} + \frac{1}{2} \delta^2 u_j^{n+1} \right], \quad \tau_j^n = \mathcal{O}\left((\Delta t)^2\right) + \mathcal{O}\left((\Delta x)^2\right), \tag{8.112}$$

where the numerical operator $\delta^2 u_j^n$ is defined as

$$\delta^2 u_j^n \equiv u_{j+1}^n - 2u_j^n + u_{j-1}^n.$$

Both of the schemes presented above are implicit schemes as both sides of their difference equations are functions of u^{n+1}, where the implicit scheme is first order in time and second order in space, while the Crank-Nicolson scheme is second order in time and space.

We return to the explicit scheme to examine its stability condition. If we consider the original continuous partial differential equation, $u_t = \sigma u_{xx}$, then the function u is a decreasing function with

time. Therefore, we would like the same property for the numerical solution. A possible solution to the differential equation is $u = e^{-k^2 t} \begin{matrix} \sin kt \\ \cos kt \end{matrix} = e^{-k^2 t} e^{ikx}$, where the term e^{ikx} is a **Fourier mode**.

Suppose that we seek a solution of the continuous partial differential equation in the form

$$u(x,t) = a(t)\, e^{ikx},$$

where $a(t) = e^{-k^2 t}$. Substituting this solution into the heat equation and attempt to imitate it with a numerical one. We start with the explicit numerical scheme;

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2} \delta^2 u_j^n.$$

Therefore, we are seeking a solution of the form $u_j^n = a_n e^{ikj\Delta x}$ for any k, where $x = j\Delta x$. We now substitute the numerical solution into the explicit numerical scheme above, which results in

$$\begin{aligned}
\frac{a_{n+1} e^{ikj\Delta x} - a_n e^{ikj\Delta x}}{\Delta t} &= \frac{\sigma}{(\Delta x)^2} \left(a_n e^{ik(j+1)\Delta x} - 2a_n e^{ikj\Delta x} + a_n e^{ik(j-1)\Delta x} \right), \\
\frac{a_{n+1} - a_n}{\Delta t} &= \frac{\sigma}{(\Delta x)^2} a_n \left(e^{ik\Delta x} - 2 + e^{-ik\Delta x} \right), \\
a_{n+1} - a_n &= \sigma \frac{\Delta t}{(\Delta x)^2} a_n \left(2\cos k\Delta x - 2 \right), \\
&= \sigma \frac{\Delta t}{(\Delta x)^2} a_n 4 \sin^2 \left(\frac{k\Delta x}{2} \right), \\
a_{n+1} &= \left(1 - 4\sigma \frac{\Delta t}{(\Delta x)^2} \sin^2 \left(\frac{k\Delta x}{2} \right) \right) a_n. \qquad (8.113)
\end{aligned}$$

Thus the expression in (8.113) can be understood as $a_{n+1} = (\text{amplitude factor})\, a_n$, and if we are to avoid the numerical method blowing up, then we require the amplitude factor to be less than or equal to one for all k. This implies the following inequality

$$\begin{aligned}
\left| 1 - 4\sigma \frac{\Delta t}{(\Delta x)^2} \sin^2 \left(\frac{k\Delta x}{2} \right) \right| &\leq 1, \qquad \forall k, \\
-1 \leq 1 - 4\sigma \frac{\Delta t}{(\Delta x)^2} \sin^2 \left(\frac{k\Delta x}{2} \right) &\leq 1, \qquad \forall k, \\
0 \leq 4\sigma \frac{\Delta t}{(\Delta x)^2} \sin^2 \left(\frac{k\Delta x}{2} \right) &\leq 2, \qquad \forall k.
\end{aligned}$$

The worse-case scenario is $\sin \frac{1}{2} k\Delta x = 1$, which results in the following conditions on Δt and Δx as

$$4\sigma \frac{\Delta t}{(\Delta x)^2} \leq 2 \Rightarrow \sigma \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}. \qquad (8.114)$$

As an example of the constraint for the explicit scheme in (8.114), if we consider the case where $\sigma = 1$, and we wish to use $\Delta x = 0.01$, then the largest value that Δt can have is $\Delta t < 0.00005$.

In Fig. 8.5 we present the *stencil* for the explicit scheme, where time is on the y-axis. As we can see from the grid, there are four different points that are involved at time $t = n\Delta t$.

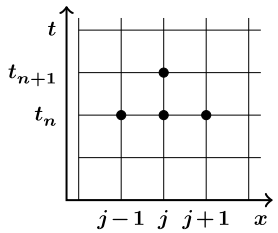


FIGURE 8.5

Illustration of the explicit forward time stencil.

We now consider the implicit scheme whose difference equation is defined as

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2}\left(u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+1}\right), \tag{8.115}$$

and address the accuracy through the truncation error and the stability through the amplification factor. The truncation error $\tau_j^n = L_h\left(u\left(x,t\right)\right)$ for the implicit scheme is

$$\tau_j^n = \frac{u\left(x_j, t_{n+1}\right) - u\left(x_j, t_n\right)}{\Delta t} - \frac{\sigma}{(\Delta x)^2}\delta^2 u\left(x_j, t_{n+1}\right). \tag{8.116}$$

As with the explicit scheme, the next step is to expand u as a Taylor series; but with respect to which expansion point? For the explicit scheme we used $u\left(x_j, t_n\right)$, which meant the left-hand side was dependent only on the expansion with respect to time, and the right-hand side was dependent only on the expansion with respect to space. However, with the implicit scheme, if we were to expand about this same point then we would have to expand $\delta^2 u\left(x_j, t_{n+1}\right)$ with respect to both t and x. Therefore, for the implicit scheme we shall expand about the point $u\left(x_j, t_{n+1}\right)$, which implies that for the left-hand side term in (8.116) we use the fact that $t_n \equiv t_{n+1} - \Delta t$.

Therefore, expanding (8.116) about $\left(x_j, t_{n+1}\right)$ results in

$$\tau_j^n = \frac{u - \left(u - \Delta t u_t + \dfrac{(\Delta t)^2}{2!}u_{tt} - \cdots\right)}{\Delta t} - \frac{\sigma}{(\Delta x)^2}\left((\Delta x)^2 u_{xx} + \frac{(\Delta x)^4}{12}u_{xxxx} + \cdots\right). \tag{8.117}$$

As with the explicit scheme the term $u_t - \sigma u_{xx}$ is the original partial differential equation and is equal to 0. Therefore, the remaining terms from (8.117) are

$$\tau_j^n = -\frac{\Delta t}{2}u_{tt} - \frac{\sigma}{12}(\Delta x)^2 u_{xxxx} + \mathcal{O}\left((\Delta t)^2\right) + \mathcal{O}\left((\Delta x)^4\right). \tag{8.118}$$

Thus the implicit scheme is consistent with the heat equation because $\Delta t \to 0$, as does Δx, with order Δt and Δx^2. This rate of convergence is the same as the explicit scheme, but we know implicit schemes involve the inversion of matrices, so if the rate of convergence is the same why would we consider the implicit scheme? To address this question, we now examine the stability of this scheme.

We again consider the Fourier stability to assess the behavior of the implicit scheme, where we assume that a true solution, of the form $u(x,t) = a(t)\,e^{ikx}$, and the numerical solution is of the form $u_j^n = a_n e^{ikj\Delta x}$; substituting the numerical solution into (8.115) results in

$$\begin{aligned}
\frac{a_{n+1}e^{ikj\Delta x} - a_n e^{ikj\Delta x}}{\Delta x} &= \frac{\sigma}{(\Delta x)^2} a_{n+1}\left(e^{ik(j+1)\Delta x} - 2e^{ikj\Delta x} + e^{ik(j-1)\Delta x}\right), \\
a_{n+1} - a_n &= \frac{\sigma\,\Delta t}{(\Delta x)^2} a_{n+1}\left(e^{ik\Delta x} - 2 + e^{-ik\Delta x}\right), \\
a_{n+1} - a_n &= \frac{\sigma\,\Delta t}{(\Delta x)^2} a_{n+1}\left(-4\sin^2\frac{1}{2}k\Delta x\right), \\
a_{n+1} &= \frac{1}{\left(1 - 4\sigma\dfrac{\Delta t}{(\Delta x)^2}\sin^2\frac{1}{2}k\Delta x\right)} a_n.
\end{aligned} \tag{8.119}$$

We now require the amplification factor in (8.119) to be less than or equal to one for all k. As we can see in (8.119), there cannot be any growth for any k. Thus the implicit scheme is **unconditionally stable** for all choices of Δt and Δx; this is why the implicit scheme is important as we do not have constraints like explicit scheme of $\sigma\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$.

8.6.3 Norms and the Maximum Principle

In Chapter 2 we introduced the concept of the vector norms, and we return to them now as they play an important part in accessing whether a numerical scheme for the partial differential equation is stable. We require a measure of *size* of u_j^n for each n. If we consider each value of u_j for $j = 0, 1, \ldots, J$ at time n, then the solution is a vector $\boldsymbol{u}^n \equiv \begin{pmatrix} u_0^n & u_1^n & \ldots & u_J^n \end{pmatrix}$. The size is measured as $\|\boldsymbol{u}^n\|$ where we have the l_2 norm ($p = 2$), l_∞ norm ($p \to \infty$) and l_1 norm ($p = 1$), and as such the general p norm is defined as

$$l_p \equiv \left(\sum_{j=0}^{J}\left(u_j^n\right)^p\right)^{1/p}.$$

Given the definition above for the different norms we have to ask the question: Which of these is equivalent to Fourier stability? **Perseval's theorem** states that the norm in *ordinary* space is the same in Fourier space. Therefore, the Fourier stability deals with the l_2 norm. However, we should note that stability depends upon the norm, thus it is possible that a scheme can be stable with respect to one norm but unstable with respect to another.

If we consider the l_∞ norm for the explicit scheme, we have

$$u_j^{n+1} = \sigma\frac{\Delta t}{(\Delta x)^2}u_{j-1}^n + \left(1 - 2\sigma\frac{\Delta t}{(\Delta x)^2}\right)u_j^n + \sigma\frac{\Delta t}{(\Delta x)^2}u_{j+1}^n,$$

and let $\mu = \sigma\dfrac{\Delta t}{(\Delta x)^2}$, then we have

$$u_j^{n+1} = \mu u_{j-1}^n + (1 - 2\mu)\,u_j^n + \mu u_{j+1}^n.$$

If $\mu \leq \frac{1}{2}$ then all the coefficients above on the right-hand side are positive, and as such we have that

$$\begin{aligned}\left|u_j^{n+1}\right| &\leq \mu\left|u_{j-1}^n\right| + (1-2\mu)\left|u_j^n\right| + \mu\left|u_{j+1}^n\right|, \\ &\leq (\mu + 1 - 2\mu + \mu)\max\left\{\left|u_{j-1}^n\right|, \left|u_j^n\right|, \left|u_{j+1}^n\right|\right\}, \\ &\leq \max\left\{\left|u_{j-1}^n\right|, \left|u_j^n\right|, \left|u_{j+1}^n\right|\right\} \leq \max_{j=0,1,\ldots,J}\left|u_j^n\right|. \end{aligned} \tag{8.120}$$

Since this is true for all j, on the left-hand side it is true for the particular j for which u_j^{n+1} is maximum over all possible js that are in the interior. This then implies that

$$\max_{j=1,2,\ldots,J-1}\left|u_j^{n+1}\right| \leq \max_{j=0,1,\ldots,J}\left|u_j^n\right|.$$

Hence

$$\begin{aligned}\max_{j=0,1,\ldots,J-1}\left|u_j^{n+1}\right| &\leq \max\left\{\max_{j=0,1,\ldots,J}\left|u_j^n\right|, \left|u_0^{n+1}\right|, \left|u_J^{n+1}\right|\right\}, \\ \Rightarrow \|\boldsymbol{u}^{n+1}\|_\infty &\leq \max\left\{\|\boldsymbol{u}^n\|_\infty, \left|u_0^{n+1}\right|, \left|u_J^{n+1}\right|\right\}.\end{aligned}$$

By successively applying this equation for $n = 0, 1, 2, \ldots$, we find that $\|\boldsymbol{u}^{n+1}\|$ is bounded by $\|\boldsymbol{u}^0\|_\infty$ and the boundary values. Therefore, the explicit scheme cannot blow up, provided $0 \leq \mu \leq \frac{1}{2}$.

If we consider the implicit scheme, then we start by stating the partial differential equation as

$$u_t = \sigma u_{xx}, \quad u(x,0) = u_0(x), \quad \begin{aligned} u(0,t) &= \phi(t), \\ u(1,t) &= \psi(t), \end{aligned}$$

where we have set the x boundaries at $x = 0$ and $x = 1$, but for the proof below, there is no loss of generality by assuming these boundaries, and the functions, ϕ and ψ, are boundary conditions for this problem. Given the implicit scheme for the heat equations

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2}\delta^2 u_j^{n+1}, \tag{8.121}$$

then we need to show that the u_j^{n+1}s are bounded. To achieve this goal we show that the maximum, but also could be minimum, value of u_j^n cannot occur in the interior of the domain shown in Fig. 8.6.

To show that the maximum, or minimum, value for u_j^n cannot occur in the interior of the domain, $[0, t] \times [0, 1]$, we shall use proof by contradiction and as such we will assume the contrary to what we wish to prove is true. Thus, we assume that u_j^n has a maximum, or minimum, in the interior of our domain at $j = j_m$ at time $n = n_m + 1$. Therefore, at this point

$$u_{j_m}^{n_m+1} > u_{j_m}^{n_m}, \tag{8.122}$$

which implies that the left-hand side of the equation in (8.121) is positive. If we now consider the right-hand side of (8.121), then we have

$$\delta^2 u_{j_m}^{n_m+1} \equiv \left(u_{j_m-1}^{n_m+1} - u_{j_m}^{n_m+1}\right) + \left(u_{j_m+1}^{n_m+1} - u_{j_m}^{n_m+1}\right). \tag{8.123}$$

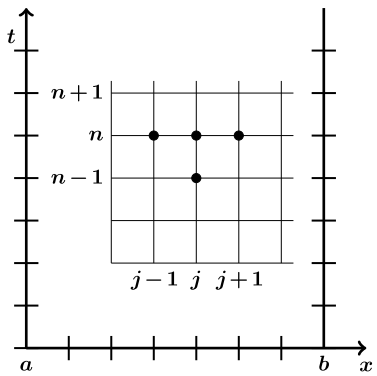


FIGURE 8.6

Illustration of the grid and stencil for the implicit upwind scheme for the proof of the maximum principle.

The important feature to note about (8.123) is that given the maximum, or minimum occurs at $u_{j_m}^{n_m+1}$, then the right-hand side of (8.123) is less than zero as both of the two terms on the right-hand side are negative. Therefore, we have a contradiction, as (8.122) implies that the left-hand side of (8.121) is positive, yet (8.123) implies that the right-hand side of (8.121) is negative. Thus this contradiction shows that u_j^n cannot have a maximum in the interior. The same argument proves that there is also no interior minimum. This implies that the maximum and minimum occur on the boundaries and as such we have

$$u_{\min} < u_j^n < u_{\max}, \quad \forall j, n,$$

which are in the interior of the domain. Thus u_j^n is contained and therefore cannot go unstable and therefore satisfies the maximum/minimum principle. This implies that the implicit scheme is unconditionally stable.

8.6.4 Implementing and Solving the Implicit Equation

We are going to consider two sets of boundary conditions for the implicit scheme: (1) where $u(0,t)$ and $u(J,t)$ are known; and (2) where $\dfrac{\partial u(0,t)}{\partial x} = 0$.

For the first case we have the boundary conditions $\begin{matrix} u_0^n = \phi_0^n \\ u_J^n = \psi_J^n \end{matrix}$. If we consider the first point inside the domain for any timestep n, we have

$$\begin{aligned} j=1: \quad & u_1^{n+1} - u_1^n = \mu u_2^{n+1} - 2\mu u_1^{n+1} + \mu u_0^{n+1}, \\ & \Rightarrow (1-2\mu)\, u_1^{n+1} - \mu u_2^{n+1} = u_1^n + \mu u_0^{n+1}, \\ j=2: \quad & u_2^{n+1} - u_2^n = \mu u_3^{n+1} - 2\mu u_2^{n+1} + \mu u_1^{n+1}, \\ & \Rightarrow -\mu u_1^{n+1} + (1-2\mu)\, u_2^{n+1} - \mu u_3^{n+1} = u_2^n, \end{aligned}$$

$$
\begin{aligned}
&\vdots \\
j = J-1: \quad & u_{J-1}^{n+1} - u_{J-1}^{n} = \mu u_{J}^{n+1} - 2\mu u_{J-1}^{n+1} + \mu u_{J-2}^{n+1}, \\
& \Rightarrow -\mu u_{J-2}^{n+1} + (1-2\mu)\, u_{J-1}^{n+1} = u_{J-1}^{n} + \mu u_{J}^{n+1}.
\end{aligned}
$$

It is possible to write the sequence above into a matrix-vector equation as

$$
\begin{pmatrix}
1+2\mu & -\mu & 0 & \cdots & \cdots & 0 \\
-\mu & 1+2\mu & -\mu & 0 & \cdots & 0 \\
0 & -\mu & 1+2\mu & -\mu & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & -\mu & 1+2\mu & -\mu \\
0 & \cdots & \cdots & 0 & -\mu & 1+2\mu
\end{pmatrix}
\begin{pmatrix}
u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_{J-2}^{n+1} \\ u_{J-1}^{n+1}
\end{pmatrix}
=
\begin{pmatrix}
u_1^{n} + \mu u_0^{n+1} \\ u_2^{n} \\ u_3^{n} \\ \vdots \\ u_{J-2}^{n} \\ u_{J-1}^{n} + \mu u_J^{n+1}
\end{pmatrix}, \tag{8.124}
$$

which is a set of $J-1$ simultaneous equations in $J-1$ unknowns so there exists a unique solution. Another feature of (8.124) is that it is a tri-diagonal matrix, where there exists fast solvers to invert these types of matrices, and so be able to solve the equations. Note that we would substitute the boundary conditions for the u_0^{n+1} and u_J^{n+1} terms in (8.124).

If we consider the second possible boundary condition, $\dfrac{\partial u\,(0,t)}{\partial x} = 0$, then we need to discretize this boundary condition as well. Using a central difference to approximate the derivative, we have $u_1^n - u_{-1}^n = 0$, but we do not know u_1^{n+1} or $u_0^{n+1} = 0$, and as such we shall include $u_1^{n+1} - u_{-1}^{n+1} = 0$ in the matrix-vector equation, which then becomes

$$
\begin{pmatrix}
-1 & 0 & 1 & \cdots & \cdots & 0 \\
-\mu & 1+2\mu & -\mu & 0 & \cdots & 0 \\
0 & -\mu & 1+2\mu & -\mu & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & -\mu & 1+2\mu & -\mu \\
0 & \cdots & \cdots & 0 & -\mu & 1+2\mu
\end{pmatrix}
\begin{pmatrix}
u_{-1}^{n+1} \\ u_0^{n+1} \\ u_1^{n+1} \\ \vdots \\ u_{J-2}^{n+1} \\ u_{J-1}^{n+1}
\end{pmatrix}
=
\begin{pmatrix}
0 \\ u_0^{n} \\ u_1^{n} \\ \vdots \\ u_{J-2}^{n} \\ u_{J-1}^{n} + \mu u_J^{n+1}
\end{pmatrix}. \tag{8.125}
$$

There are different types of boundary conditions that could be given:

- *Dirichlet*: Values of u given all round the boundary.
- *Neumann*: Values of the normal derivative $\dfrac{\partial u}{\partial n}$ are given on the boundary.
- *Robin*: A mixed combination of Dirichlet and Neumann $\alpha \dfrac{\partial u}{\partial n} + \beta u = 0$.

8.6.5 θ-Methods

The three schemes presented so far to numerically solve the heat equation are of the class of problems referred to as θ-methods. They are defined as $(1-\theta)$ (explicit) $+\,\theta$(implicit). Therefore, when $\theta = 0$

we have the explicit scheme that we have already considered, and when $\theta = 1$ we have the fully implicit scheme.

If we consider the general expression for the θ-method, then we have

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2}\left[(1-\theta)\,\delta^2 u_j^n + \theta\delta^2 u_j^{n+1}\right], \quad 0 \le \theta \le 1. \tag{8.126}$$

We now consider the Fourier stability of (8.126). As before we assume that the numerical solution is of the form $u_j = a_n e^{ikj\Delta x}$ and substitute this expression into (8.126), which, after cancelations, becomes

$$\begin{aligned} a_{n+1} - a_n &= \mu\left[(1-\theta)\left(-4\sin^2\frac{1}{2}k\Delta x\right)a_n + \theta\left(-4\sin^2\frac{1}{2}k\Delta x\right)a_{n+1}\right], \\ &= -4\mu\sin^2\frac{1}{2}k\Delta x\left[(1-\theta)\,a_n + \theta a_{n+1}\right], \\ \Rightarrow a_{n+1} &= \left(\frac{1 - 4(1-\theta)\,\mu\sin^2\frac{1}{2}k\Delta x}{1 + 4\theta\mu\sin^2\frac{1}{2}k\Delta x}\right)a_n. \end{aligned} \tag{8.127}$$

For the θ-method to be stable, we require the amplification factor in (8.127) to be less than or equal to one in modulus. Therefore, we have the inequality

$$-\left(1 + 4\theta\mu\sin^2\frac{1}{2}k\Delta x\right) \le 1 - 4\mu(1-\theta)\sin^2\frac{1}{2}k\Delta x \le 1 + 4\mu\sin^2\frac{1}{2}k\Delta x. \tag{8.128}$$

The right-hand side of the inequality in (8.128) is satisfied for all values of θ, so we consider the left-hand side of the inequality to see if there are any constraints on θ or μ. Rearranging (8.128) so that the terms involving θ are on the same side of the inequality sign, we obtain

$$4\mu\sin^2\frac{1}{2}k\Delta x \le 2. \tag{8.129}$$

If $\theta \ge \frac{1}{2}$ the inequality in (8.129) is always satisfied. Otherwise we require

$$\mu \le \frac{1}{2(1-2\theta)}, \tag{8.130}$$

where we have taken the worse case scenario that $\sin^2\frac{1}{2}k\Delta x = 1$, which occurs when $\frac{1}{2}k\Delta x = m\pi$.

We now consider the case where $\theta = \frac{1}{2}$, which is what we have referred to as the **Crank-Nicolson** scheme. Therefore, given the theory that we have just derived, we know that the Crank-Nicolson scheme is unconditionally stable. The next step is to verify if the Crank-Nicolson scheme is consistent with the heat equation. Therefore, formulating the truncation error, we have

$$\begin{aligned} \tau_j^n = \frac{u(x,t+\Delta t) - u(x,t)}{\Delta t} &- \frac{\sigma}{2(\Delta x)^2}\left(u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)\right. \\ &\left. + u(x+\Delta x,t+\Delta t) - 2u(x,t+\Delta t) + u(x-\Delta x,t+\Delta t)\right). \end{aligned} \tag{8.131}$$

We now apply a Taylor series expansion in two variables where we shall expand about the point x, $t+\frac{1}{2}\Delta t$. This then makes (8.131)

$$\begin{aligned}\tau_j^n = &\frac{\left(u+\frac{1}{2}\Delta t u_t+\frac{1}{2}\left(\frac{\Delta t}{2}\right)^2 u_{tt}+\cdots\right)-\left(u-\frac{1}{2}\Delta t u_t+\frac{1}{2}\left(\frac{\Delta t}{2}\right)^2 u_{tt}+\cdots\right)}{\Delta t}\\ &-\frac{\sigma}{2(\Delta x)^2}\left((\Delta x)^2 u_{xx}+(\Delta x)^2\frac{\Delta t}{2}u_{xxt}+\cdots+\frac{(\Delta x)^4}{12}u_{xxxx}+(\Delta x)^4\frac{\Delta t}{2}u_{xxxxt}+\cdots\right.\\ &\left.+(\Delta x)^2 u_{xx}-(\Delta x)^2\frac{\Delta t}{2}u_{xxt}+\cdots+\frac{(\Delta x)^4}{12}u_{xxxx}-(\Delta x)^4\frac{\Delta t}{2}u_{xxxxt}+\cdots\right),\\ &=\frac{1}{12}(\Delta t)^2 u_{ttt}-\sigma\frac{(\Delta x)^2}{12}u_{xxxx}+\cdots.\end{aligned}$$

Therefore, we have just proven that the Crank-Nicolson scheme is consistent with the heat equation and that it is accurate to second order in both time and space. Thus Crank-Nicolson scheme has a higher-order accuracy in time than both the explicit and the implicit schemes.

Given all the derivations that we have shown so far for the heat equations, it is now possible to put together a convergence theorem for the heat equation. Let $u_t=\sigma u_{xx}$ and $\frac{u_j^{n+1}-u_j^n}{\Delta t}=\frac{\sigma}{(\Delta x)^2}\delta^2 u_j^n$ with

$$\tau_j^n=\frac{u\left(t_{n+1},x_j\right)-u\left(t_n,x_j\right)}{\Delta t}-\frac{\sigma}{(\Delta x)^2}\delta^2 u\left(t_n,x_j\right), \tag{8.132}$$

$$0=\frac{u_j^{n+1}-u_j^n}{\Delta t}-\frac{\sigma}{(\Delta x)^2}\delta^2 u_j^n. \tag{8.133}$$

Next, let the error E_j^n be defined as $E_j^n\equiv u\left(t_n,x_j\right)-u_j^n$. Subtracting (8.133) from (8.132) yields

$$\tau_j^n=\frac{E_j^{n+1}-E_j^n}{\Delta t}-\frac{\sigma}{(\Delta x)^2}\delta^2 E_j^n. \tag{8.134}$$

Rearranging (8.134) to isolate E_j^{n+1} and expand the numerical difference operator δ^2, we obtain

$$E_j^{n+1}=\mu E_{j-1}^n+(1-\mu)E_j^n+\mu E_{j+1}^n+\Delta t\tau_j^n, \tag{8.135}$$

where $\mu\equiv\frac{\sigma}{(\Delta x)^2}$.

Taking the modulus of (8.135) results in

$$\left|E_j^{n+1}\right|\le|\mu|\left|E_{j-1}^n\right|+|(1-\mu)|\left|E_j^n\right|+|\mu|\left|E_{j+1}^n\right|+\Delta t\left|\tau_j^n\right|. \tag{8.136}$$

Let $\|E^n\|=\max_j\left|E_j^n\right|$ and $\|\tau^n\|=\max_j\left|\tau_j^n\right|$, then we have an upper bound for the error on the left-hand side of the inequality in (8.136) as

$$\left|E_j^{n+1}\right|\le(|\mu|+|(1-2\mu)|+|\mu|)\|E^n\|+\Delta t\|\tau^n\|. \tag{8.137}$$

If we now assume that $\mu \leq \frac{1}{2}$, then we obtain

$$\left|E_j^{n+1}\right| \leq \|E^n\| + \Delta t \|\tau^n\|. \tag{8.138}$$

Since (8.138) is true for all j, then in particular that j for which $\left|E_j^{n+1}\right| = \max_j \left|E_j^{n+1}\right|$, thus we have

$$\|E^{n+1}\| \leq \|E^n\| + \Delta t \|\tau^n\|.$$

Solving for $\|E^n\|$, we have

$$\begin{aligned}\|E^n\| &\leq \|E^0\| + \Delta t \sum_{m=0}^{n-1} \|\tau^m\|, \\ &\leq \|E^0\| + n\Delta t \max_{m=0,\ldots,n-1} \|\tau^m\|.\end{aligned}$$

We now make the assumption that the norm of the error at the initial time is zero; this is consistent with knowing the exact initial conditions which we have been given. Therefore we now have

$$\|E^n\| \leq \|E^0\| + t \max_{m=0,\ldots,n-1} \|\tau^m\|. \tag{8.139}$$

Since the truncation error tends to zero as Δt and Δx both tend to zero, we then have that $\|E^n\| \to 0$, provided $\sigma \frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$ and $\|E^0\| = 0$.

In conclusion, we have shown that if a scheme is consistent with respect to the partial differential equations, and is a stable numerical approximation, then the numerical scheme will converge to the true solution to the heat equation as both the temporal and spatial step sizes tend to zero. However, we have only proved the convergence of one numerical schemes for the heat equations so far. We have shown that there is a constraint on the size of the ratio of Δt and Δx that they must satisfy $\mu \leq \frac{1}{2}$ for stability of the numerical scheme. It is possible to rearrange this inequality so that given a value for Δx, then we can determine which values Δt can take, that is to say $\Delta t = \frac{(\Delta x)^2}{2\sigma}$. It is then possible to define a path in Δx and Δt space which characterize convergence. This boundary in the Δx and Δt space is referred to as the **refinement path**, where if the ratio of the step sizes falls on one side then the numerical scheme will converge to the true solution, but will not do so if the ratio is on the other side.

8.6.6 More Generous Stability Condition

When we were considering the ordinary differential equation example $y' = \lambda y$, the numerical scheme had an associated stability polynomial. For absolute stability we had the condition $Z_r \in x^2 + y^2 \leq 1$, for the situations where $\lambda \leq 0$ and as such the true solution is decaying or is a constant, but what happens when $\lambda > 0$? In this situation we have that the analytical solution is growing, and as such we require our numerical solution also to grow but not to blow up.

To allow for some growth in the numerical approximation, we have

$$y = y_0 e^{\lambda t} = y_0 e^{\lambda n h} = y_0 \left(e^{\lambda h}\right)^n = y_0 Z^n, \tag{8.140}$$

where we have $Z = e^{\lambda h}$. Thus the new stability condition is $|Z_r| \leq e^{\lambda h}$.

If we consider $|Z|^n \leq K$ for $nh = t$, $K = e^{\ln K}$, then

$$K^{\frac{1}{n}} = e^{\frac{1}{n}\ln K} = e^{h\frac{\ln K}{t}} = 1 + \frac{h\ln K}{t} + \mathcal{O}\left(h^2\right) = 1 + \mathcal{O}(h). \tag{8.141}$$

This new condition is referred to as the **Von Neumann stability condition**, which allows extra leeway in the behavior of the numerical scheme. This extra leeway also extends to partial differential equations as well, where it becomes $|Z| = 1 + \mathcal{O}(\Delta t)$.

As an example, let us consider the extended heat equation

$$u_t = \sigma u_{xx} + u, \tag{8.142}$$

and apply the explicit scheme derived earlier:

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \frac{\sigma}{(\Delta x)^2}\delta^2 u_j^n + u_j^n. \tag{8.143}$$

Applying the Fourier stability approach $u_j^n = a_n e^{ikj\Delta x}$ to (8.143), we obtain

$$\begin{aligned}
\frac{a_{n+1}e^{ikj\Delta x} - a_n e^{ikj\Delta x}}{\Delta t} &= \frac{\sigma}{(\Delta x)^2}\left(a_n e^{ik(j-1)\Delta x} - 2a_n e^{ikj\Delta x} + a_n e^{ik(j-1)\Delta x}\right) + a_n e^{ikj\Delta x}, \\
a_{n+1} - a_n &= \frac{\sigma}{(\Delta x)^2}\left(-4\sin^2\frac{1}{2}k\Delta x\right)a_n + \Delta t a_n, \\
a_{n+1} &= \left(1 - 4\mu\sin^2\frac{1}{2}k\Delta x + \Delta t\right)a_n.
\end{aligned} \tag{8.144}$$

The condition in (8.144) is of the Von Neumann stability type $1 + \mathcal{O}(\Delta t)$ and as such we focus on the first term, where we still have the condition that $\mu \leq \frac{1}{2}$, and as such this defines the refinement path to ensure convergence.

Exercise 8.6. *Determine the order of the accuracy and the stability of the following three schemes for approximating the heat equation $u_t = u_{xx}$:*

1. $\dfrac{1}{2\Delta t}\left(u_j^{n+1} - u_j^{n-1}\right) = \dfrac{1}{(\Delta x)^2}\delta^2 u_j^n,$
2. $\dfrac{1}{2\Delta t}\left(u_j^{n+1} - u_j^{n-1}\right) = \dfrac{1}{(\Delta x)^2}\left(u_{j-1}^n + u_{j+1}^n - u_j^{n+1} - u_j^{n-1}\right),$
3. $\dfrac{3}{2\Delta t}\left(u_j^{n+1} - u_j^n\right) - \dfrac{1}{2\Delta t}\left(u_j^n - u_j^{n-1}\right) = \dfrac{1}{(\Delta x)^2}\delta^2 u_j^n,$

where $\delta^2 u_j^n \equiv u_{j+1}^n - 2u_j^n + u_{j-1}^n$.

8.7 Wave Equation

As we showed at the start of the partial differential equation section, the wave equation can be either a second-order partial differential equation $u_{tt} = c^2 u_{xx}$, or it can be a first-order partial differential

equation of the form $u_t + cu_x = 0$. In this section we shall consider the second-order partial differential equation, but we consider the system of first-order partial differential equations which arise from $u_t = cv_x$, $v_t = cu_x$, and can be written as a system of partial differential equations:

$$\begin{pmatrix} u \\ v \end{pmatrix}_t = \begin{pmatrix} 0 & c \\ c & 0 \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x \Rightarrow \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} p \\ q \end{pmatrix} \Rightarrow \begin{array}{l} p_t - cp_x = 0 \\ q_t + cq_x = 0. \end{array} \tag{8.145}$$

8.7.1 Forward-Time, Centered-Space

Therefore, we are going to consider schemes for $u_t + cu_x = 0$. The first scheme is referred to as a **central difference** scheme for space; the actual full numerical scheme is referred to as forward-time, centered-space and is defined as

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = c\frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} = 0, \tag{8.146}$$

and is first order in Δt and second order in Δx. However, if we consider the stability analysis through the Fourier method where we substitute $u_j^n = a_n e^{ikj\Delta x}$ into (8.146), then we see that

$$\begin{aligned} a_{n+1} - a_n + c\frac{\Delta t}{2\Delta x}\left(e^{ik\Delta x} - e^{-ik\Delta x}\right)a_n = 0, \\ a_{n+1} = \left(1 - \frac{c\Delta t}{\Delta x} i \sin k\Delta x\right) a_n. \end{aligned} \tag{8.147}$$

There is an important feature to note about the amplification factor in (8.147) is that it is not inside or on unit circle for all k. Therefore, the scheme is not absolute stable. This numerical approximation is actually unstable, although we should note here that the scheme is not unstable straight away but will do so over time no matter what sizes the spatial and temporal step sizes take. Thus this scheme is useless and should be avoided for long-term integrations.

8.7.2 Explicit Upwind

The next approach that we shall consider is referred to as an **upwind scheme** and is used when $c > 0$. The difference equations for the upwind scheme is given by

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + \frac{c}{\Delta x}\left(u_j^n - u_{j-1}^n\right) = 0. \tag{8.148}$$

The upwind scheme in (8.148) is first order in Δt and in Δx. Again forming the Fourier stability with $u_j^n = a_n e^{ikj\Delta x}$, we obtain

$$\begin{aligned} a_{n+1} - a_n + \frac{c\Delta t}{\Delta x}\left(1 - e^{ik\Delta x}\right)a_n = 0, \\ a_{n+1} = \left\{1 - \frac{c\Delta t}{\Delta x}\left(1 - e^{ik\Delta x}\right)\right\} a_n. \end{aligned} \tag{8.149}$$

Therefore, the amplification factor for the upwind scheme is $1 - \nu + \nu e^{ik\Delta x}$, where $\nu \equiv \frac{c\Delta t}{\Delta x}$. We thus have a real and a complex component to the amplitude factor

$$a_{n+1} = \left(\underbrace{(1-\nu)}_{\mathbb{R}} + \underbrace{\nu e^{-ik\Delta x}}_{\mathbb{C}} \right) a_n.$$

Therefore, we have a circle of radius ν inside the unit circle such that the upwind scheme is stable. This is true of $\nu < 1$. The circle will go outside the unit circle if $\nu \geq 1$, which would then make the scheme unstable. Therefore, the upwind scheme is conditionally stable.

8.7.3 Implicit Upwind

The next scheme we consider is the **implicit upwind scheme**, which is defined as

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} + c\frac{u_j^{n+1} - u_{j-1}^{n-1}}{\Delta x} = 0. \tag{8.150}$$

Again applying the Fourier stability analysis with $u_j^n = a_n e^{ikj\Delta x}$, it can easily be shown that

$$\left(1 + \frac{c\Delta t}{\Delta x}\left(1 - e^{-k\Delta x}\right)\right) a_{n+1} = a_n. \tag{8.151}$$

Therefore, the amplification factor is

$$a_{n+1} = \frac{1}{1 + \frac{c\Delta t}{\Delta x}\left(1 - e^{-ik\Delta x}\right)} a_n \equiv \left(\frac{1}{1 + \nu - \nu e^{ik\Delta x}}\right) a_n. \tag{8.152}$$

As with the explicit upwind scheme, the amplitude factor for the implicit scheme also consists of a real and complex component. The real component is a circle centered at $1 + \nu$ which is outside the unit circle; however, as we require the reciprocal of this circle, that reciprocal circle lies within the unit circle, which then makes the implicit upwind scheme unconditionally stable.

8.7.4 Box Scheme

The next scheme that we consider is the **box scheme**, which uses a Crank-Nicolson type approach but for the first-order derivatives, therefore we are using

$$u_t \approx \frac{1}{2}\frac{\left(u_j^{n+1} - u_j^n\right)}{\Delta t} + \frac{1}{2}\frac{\left(u_{j+1}^{n+1} - u_{j+1}^n\right)}{\Delta t},$$
$$u_x \approx \frac{1}{2}\frac{\left(u_{j+1}^n - u_j^n\right)}{\Delta x} + \frac{1}{2}\frac{\left(u_{j+1}^{n+1} - u_j^{n+1}\right)}{\Delta x}.$$

The truncation error for the box scheme can be shown to be $\tau_j^n = \mathcal{O}\left((\Delta x)^2\right) + \mathcal{O}\left((\Delta t)^2\right)$.

For the stability of the box scheme, we again apply the Fourier stability analysis $u_j^n = a_n e^{ikj\Delta x}$, which yields

$$\begin{aligned}
&\frac{1}{2\Delta t}\left(a_{n+1}-a_n\right)e^{ikj\Delta x}+\frac{1}{2\Delta t}\left(a_{n+1}-a_n\right)e^{ik\overline{j+1}\Delta x}\\
&\quad+c\left(\frac{1}{2\Delta x}\left(e^{ik\overline{j+1}\delta x}-e^{ikj\Delta x}\right)a_n+\frac{1}{2\Delta x}\left(e^{ik\overline{j+1}\delta x}-e^{ikj\Delta x}\right)a_{n+1}\right)=0,\\
&\Rightarrow\left(a_{n+1}-a_n\right)\left(1+e^{ik\Delta x}\right)+\nu\left(a_n+a_{n+1}\right)\left(e^{ik\Delta x}-1\right)=0,\\
&\Rightarrow\left(a_{n+1}-a_n\right)\cos\frac{1}{2}k\Delta x+\nu\left(a_n+a_{n+1}\right)i\sin\frac{1}{2}k\Delta x=0,\\
&\Rightarrow\frac{a_{n+1}}{a_n}=\frac{\cos\frac{1}{2}k\Delta x-i\nu\sin\frac{1}{2}k\Delta x}{\cos\frac{1}{2}k\Delta x+i\nu\sin\frac{1}{2}k\Delta x}\Rightarrow\left|\frac{a_{n+1}}{a_n}\right|=1,\quad\forall\nu.
\end{aligned}\tag{8.153}$$

Thus the box scheme is unconditionally stable for all choices of Δx and Δt, but it is an implicit scheme which implies the inversion of a matrix to obtain the numerical approximation.

8.7.5 Lax-Wendroff Scheme

The Lax-Wendroff scheme is derived from considering a Taylor series expansion of $u(x,t)$ about the time component, which is

$$u(x,t+\Delta t)=u(x,t)+\Delta t u_t(x,t)+\frac{1}{2}(\Delta t)^2 u_{tt}(x,t)+\cdots.$$

The next step is to use the property of the first-order version of the wave equation that $u_t=-cu_x$. Substituting this property into the equation above results in

$$u(x,t+\Delta t)=u(x,t)-c\Delta t u_x(x,t)+\frac{1}{2}(\Delta t)^2 u_{tt}(x,t)+\cdots.$$

Next, we recall the original second-order version of the wave equation, $u_{tt}=c^2u_{xx}$, and substitute this into the second-order term above, which results in

$$u(x,t+\Delta t)=u(x,t)-c\Delta t u_x(x,t)+c^2\frac{1}{2}(\Delta t)^2 u_{xx}(x,t)+\cdots.\tag{8.154}$$

The difference equation for the Lax-Wendroff scheme comes about through considering central differences in the space component for the first- and second-order spatial derivatives in (8.154), which is given by

$$u_j^{n+1}=u_j^n-\frac{c\Delta t}{2\Delta x}\left(u_{j+1}^n-u_{j-1}^n\right)+\frac{c^2(\Delta t)^2}{2(\Delta x)^2}\left(u_{j+1}^n-2u_j^n+u_{j-1}^n\right).\tag{8.155}$$

As we can see from (8.155), the Lax-Wendroff scheme is an explicit numerical approximation to the wave equation. It can be shown that the Lax-Wendroff scheme has a truncation error $\tau_j^n=\mathcal{O}\left((\Delta t)^2\right)+\left((\Delta x)^2\right)$, which implies that the Lax-Wendroff scheme is second order in space and time. To determine

if there are any constraints on the spatial and temporal step sizes we again consider the Fourier stability of (8.155) through substituting $u_j^n = a_n e^{ikj\Delta x}$ into (8.155), which yields

$$a_{n+1} = \left(1 - \nu^2 + \nu^2 \cos k\Delta x - i\nu \sin k\Delta x\right) a_n. \tag{8.156}$$

The amplification factor in (8.156) lies on an ellipse that is centered at $\left(1 - \nu^2, 0\right)$ with semiaxes $\left(\nu^2, -\nu\right)$, which lies inside the unit circle if $|\nu| \leq 1$.

8.8 Courant Friedrichs Lewy Condition

The behavior of partial differential equations is govern by their characteristics. The characteristics for the first-order wave equation $u_t + cu_x = 0$ depend upon the line from that has slope c^{-1}, and as such the solution depends only on values that are on that line. This is referred to as the **domain of dependence**. The numerical approximations to the wave equation depend on values at the stencil points being used to approximate the partial differential equation at time level n. This is referred to as the **numerical domain of dependence**.

If the line does not lie in the numerical domain of dependence, then the numerical solution and the exact solution may depend on completely different initial values, so convergence cannot take place. This is the Courant Friedrichs Lewy (CFL) condition, or more commonly referred to as the CFL condition. The CFL condition is a necessary condition for convergence. If we consider the central difference and the Lax-Wendroff schemes then their CFL conditions are $|\nu| \leq 1$. For the explicit upwind scheme the CFL condition $0 \leq \nu \leq 1$. The slope of the characteristic of the wave equation has a slope of c^{-1}. We have $c\Delta t \leq \Delta x \Rightarrow \nu \leq 1$. This then leads to the following theorem.

Theorem 8.7. ***The lax equivalence theorem****: For a* ***well-posed*** *initial value problem and a* ***consistent*** *difference approximation, then convergence is equivalent to stability.*

8.9 Summary

In this chapter we have introduced the theory of numerical approximations to initial value problems. We have considered both ordinary and partial differential equations. We have introduced the concepts of consistency, stability and convergence and identified the constraints that could apply to the scheme that you may be considering. For the ordinary differential equations we have been able to derive general properties for the different discretizations, but for the partial differential equations, the properties are equation- and scheme-specific.

Numerical modeling plays a vital part in all forms of data assimilation and in predictability in all of the geosciences. The reason for this first of six chapters on numerical modeling, with boundary value problems following on in the next chapter, is to make the reader aware of why the numerical scheme of choice works, but also how to assess if something is wrong if the scheme has an adaptive step size. It is possible to have CFL conditions that can be dynamically dependent, and as such it could be that the data assimilation scheme analysis could be a state that is not inside the domain of dependence or forces the amplification factor outside the unit circle, but is not within the Neumann stability region for growing solutions.

CHAPTER

Numerical Solutions to Boundary Value Problems

9

Contents

In this chapter we shall extend the theory developed in the previous one for initial value ordinary and partial differential equations to the class of boundary value differential equations. We shall develop various numerical approximations to different forms of differential equations, show that they are consistent, as well as derive bounds for the associated errors. We start be defining the two different forms of differential equations that we considered over the last chapter and in this one.

9.1 Types of Differential Equations

In the last chapter we considered **initial value problems**, where the ordinary differential equation was of the form

$$y' + y = 0, \quad y(0) = 1, \quad \Rightarrow y = e^{-x}, \tag{9.1}$$

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00013-9

with the operator, $'$, being a differential operator. For the initial value problem this operator was associated with time, but it could also be associated with derivative of the spatial coordinates: x, y, or z. Initial value ordinary differential equations are problems where the conditions are given only at one point. A second-order example on an initial value differential equation is

$$y'' + y' = 0, \quad y(0) = 1, \quad y'(0) = 0, \tag{9.2}$$

where in (9.2) we have two initial conditions: the first condition is for the solution itself, while the second condition is on the first derivative of the solution.

In this chapter we consider numerical approximations to **boundary value problems**, where the information is given at two or more points for a nth-order differential equation for $n \geq 2$. A second-order example of a boundary value problem is

$$y'' + y = 0, \quad y(0) = 1, \quad y(1) = 0, \quad \left(y(0) = 1,\ y'(1) = 1\right), \tag{9.3}$$

where we have presented two possible types of boundary conditions in (9.3). A fourth-order example of a boundary value problem is

$$y'''' + y = 0, \quad y(0) = a, \quad y'(0) = b, \quad y''(1) = c, \quad y(1) = d. \tag{9.4}$$

The next set of problems that we consider are referred to as **eigenvalue problems**. An example of this type of problem, and its analytical solution derivation, is

$$\begin{aligned} &y'' + \omega^2 y = 0, \quad x \in [0, 1], \quad y(0) = y(1) = 0, \\ &\quad \Rightarrow y = A\cos\omega x + B\sin\omega x, \\ &\quad x = 0 \Rightarrow A = 0, \\ &\quad x = 1 \Rightarrow B\sin\omega = 0. \end{aligned}$$

We do not wish to have a zero solution for the problem above, so to obtain a non-trivial solution we require $\omega = n\pi$, where $n \in \mathbb{Z}$ and $\mathbb{Z}$ is the set of positive integers, which gives the solution as $y = B\sin(k\pi x)$, where $k \in \mathbb{Z}$ but $B \neq 0$ but is still undetermined.

The final set of problems that we consider are the **partial differential equations**, where the heat equation in the previous chapter is an example, where now we have boundary conditions

$$u_t = u_{xx} \quad \begin{matrix} u(x, 0) = f(x) \\ u_t(x, 0) = g(x) \end{matrix}, \quad u(0, t) = u(1, t) = 0. \tag{9.5}$$

Boundary value partial differential equations are elliptical differential equations, which means that there is a condition for the continuous solution to the differential equation to exist. We shall go into more detail about the *ellipticity condition* in Chapter 17. Another frequently used partial differential equation in geosciences is the **Laplace equation**, which is given by

$$\nabla^2 u \equiv u_{xx} + u_{yy} = 0. \tag{9.6}$$

When the right-hand side of (9.6) is non-zero, this differential equation is referred to as a **Poisson equation**, which is given by

$$\nabla^2 u = f. \tag{9.7}$$

Both the Laplace and the Poisson equations could be defined in a domain denoted $\mathcal{D}$, where the boundary of the domain is denoted by $\partial\mathcal{D}$. We have presented a simple example of the domain in Fig. 9.1. For boundary value problems we would have information on $\partial\mathcal{D}$, which could be in the form $u = f$ on $\partial\mathcal{D}$, which is a Dirichlet condition, or it could be $\frac{\partial u}{\partial n} = g$ on $\partial\mathcal{D}$, which are referred to as Neumann conditions.

A higher-order differential equation is the biharmonic partial differential equation, given by

$$\nabla^4 u = u_{xxxx} + u_{yyyy} = 0, \tag{9.8}$$

where the boundary conditions u and $\frac{\partial u}{\partial n}$ could be given or $\nabla^2 = u$ could be given.

We now return to the eigenvalue problem $y'' + y = 0$, where there is much known about the existence and uniqueness of the solution to this type of problem when it is in the form of an initial value problem. We now consider the case where we have boundary conditions instead of only initial conditions:

$$y'' + y = 0, \quad x \in [0, l], \qquad \begin{array}{l} y(0) = 0, \\ y(l) = y_l. \end{array} \tag{9.9}$$

We know that the solution to (9.9) is $y = A\cos x + B\sin x$. However, with the first boundary condition we arrive at $A = 0$. When we consider the second boundary condition at $x = l$, we obtain $b = \frac{y_l}{\sin l}$ as long as $\sin l \neq 0$. However, it is possible for $\sin l$ to be equal to zero and we have to consider these situations. (1) If $\sin l = 0$, which occurs at $l = n\pi$ and if $y_l = 0$, then the constant B is undetermined. (2) However, if $\sin l = 0$ for $l = n\pi$ but $y_l \neq 0$, then there is no solution to this differential equation. In summary, we have

1. **uniqueness** when $\sin l \neq 0$ and $y = \frac{y_l}{\sin l}\sin x$;
2. **no solution** when $\sin l = 0$, where $l = n\pi$ and $y_l \neq 0$; and
3. **infinity of solutions** when $\sin l = 0$, where $l = n\pi$ and $y_l = 0$, as $y = B\sin x$ for any B.

In this chapter we consider two different numerical approaches for solving boundary value problems: the **shooting method** and **finite differences method**. The latter was the technique used for the initial value problems in the previous chapter.

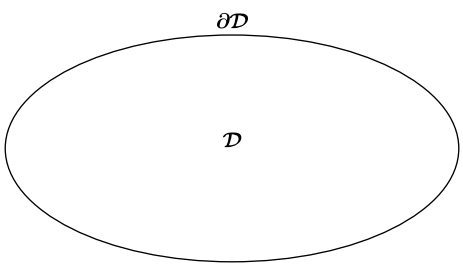


FIGURE 9.1

Diagram of a possible domain and its boundary for the Laplace or the Poisson equation.

9.2 Shooting Methods

We first define the standard second-order boundary value problem as

$$\begin{aligned} y'' &= f\left(x, y, y'\right), \quad x \in [a, b], \\ y(a) &= \alpha, \\ y(b) &= \beta, \end{aligned} \tag{9.10}$$

where a, b, α, and β are given constants. We also consider the associated second-order initial value problem defined as

$$\begin{aligned} y'' &= f\left(x, y, y'\right), \quad x \in [a, b], \\ y(a) &= \alpha, \\ y'(a) &= s. \end{aligned} \tag{9.11}$$

The solution of the initial value problem (9.11) is denoted as $y(x, s)$ and the aim of the shooting method is to find a $s = s^*$ for which the solution of (9.11) is the same as the solution of (9.10). That is to say, $y(b, s^*) = \beta$.

An important thing to note here is that we shall solve (9.11) by using a numerical integration procedure, for example, the Euler method for initial value problems. However, this means that it is necessary to rewrite (9.11) as a system of first-order differential equations. Thus let $y' = v$, which implies $v' = y'' = f(x, y, v)$. Rewriting the initial conditions in terms of v, we have $y(a) = \alpha$ and $v(a) = y'(a) = s$.

If we were to solve (9.11), then we would obtain a solution in the form $y(x, s)$, with the function $g(s) = y(b, s) - \beta$, with the property $g(s^*) = 0$; however, this results in a nonlinear equation to solve. Another important thing to notice here is that we shall not, in practice, find $y(x, s)$, but only a numerical approximation to the solution. We therefore have to consider which methods are applicable for determining s^*.

1. A so-called ad hoc, or trial and error, approach is where we would choose an s_1, determine $y(x, s_1)$, observe $y(b, s_1)$, then modify s_1, and repeat the steps.
2. We could use a *bisection* method that is solved with initial values for s, s_1, and s_2, where $g(s_1 \mid g(s_2)) < 0$ and then let $s_3 = \frac{1}{2}(s_1 + s_2)$, then determine $g(s_3)$ and compare $g(s_1)$ with $g(s_3)$ and $g(s_2)$ with $g(s_3)$ to see which one is closest, in order to determine where the bisection will occur.
3. Linear interpolation.

To demonstrate how the linear interpolation approach would be applied, we have presented an illustration in Fig. 9.2.

We can see in this figure, the value for s^* that enables $g(s^*) = y(b, s^*) = \beta$, lies at some point between s_1 and s_2, where the values of the function y is known; therefore, to build the linear interpolation we have

$$\frac{y(b, s_2) - y(b, s_1)}{s_2 - s_1} = \frac{y\left(b, s^*\right) - y(b, s_1)}{s^* - s_1}. \tag{9.12}$$

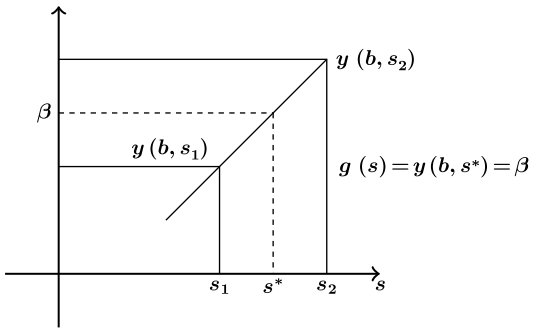


FIGURE 9.2

Diagram of the setup for the linear interpolation scheme for the shooting method.

Rearranging (9.12) to isolate s^*, and using the property that $y\,(b, s^*) = \beta$, we obtain

$$\begin{aligned} s^* &= s_1 + \frac{(s_2 - s_1)\,(\beta - y\,(b, s_1))}{y\,(b, s_2) - y\,(b, s_1)}, \\ &\equiv s_1 - \frac{g\,(s_1)}{(g\,(s_2) - g\,(s_1))\,/\,(s_1 - s_2)}. \end{aligned} \tag{9.13}$$

In general the linear interpolation approach defined in (9.13) only produces a better approximation to s^* and not the exact value. The expressions in (9.13) is equivalent to the **Secant method**, which is an approximation to the **Newton method**.

For the linear problem we have equivalent general second-order boundary value problem given by

$$y'' + py' + qy = r, \quad x \in [a, b], \quad \begin{matrix} y\,(a) = \alpha \\ y\,(b) = \beta \end{matrix}, \tag{9.14}$$

and the associated linear initial value problem is

$$y'' + py' + qy = r, \quad x \in [a, b], \quad \begin{matrix} y\,(a) = \alpha \\ y'\,(a) = s \end{matrix}. \tag{9.15}$$

We can then form the equivalent linear system of partial differential equations as

$$\begin{aligned} y' &= v, \\ v' &= r - pv - qy, \end{aligned}$$

with the initial condition $y\,(a) = \alpha$. Therefore, solving the initial value problem from (9.15) gives $y\,(x, s)$ and our aim is to find $s = s^*$ such that $g\,(s^*) = 0$, where $g\,(s) = y\,(b, s) - \beta$. Thus, in this case the linear interpolation immediately gives s^*.

To help illustrate this property of the linear interpolation for the linear shooting method, we consider Example 9.1.

Example 9.1. *Consider the second-order linear boundary value problem*

$$y'' + y' = 1, \quad x \in [0,1], \quad \begin{matrix} y(0) = 0 \\ y(1) = 1 \end{matrix}. \tag{9.16}$$

State the associated linear initial value problem and show that the shooting method using linear interpolation finds the solution in one step.

Solution. The general solution to (9.16) is given by $y = Ae^{-x} + B + x$, but when applying the boundary conditions' results in $y = x$. The associated linear interpolation problem for (9.16) is

$$y'' + y' = 1, \quad x \in [x,b], \quad \begin{matrix} y(0) = 0 \\ y'(0) = s \end{matrix}. \tag{9.17}$$

As shown for the boundary value problem, the solution to the initial value problem in (9.17) is $y = ce^{-x} + d + x$. Applying the first initial condition, we have $0 = C + D$, and then the second initial condition gives $y' = -ce^{-x} + 1$, then $s = -c + 1 \Rightarrow c = 1 - s$, and $D = s - 1$. This forms $y(x,s) = (1-s)e^{-x} + s - 1 + x$ and $y(1,s^*) = (1-s^*)e^{-1} + (s^* - 1) + 1 = 1$, implying $s^* = 1$.

Now, forming the linear interpolation approach, we have $y(1,s) = (1-s)e^{-1} + s - 1 + 1 = (1-s)e^{-1}$. Setting $s = s_1 = 0$, we have $y(1,s_1) = e^{-1}$ and $s = s_2 = 2$, which results in $y(1,s_2) = -e^{-1} + 2$. Finally, forming s^*, we have

$$s^* = 0 + \frac{(2-0)\left(1 - e^{-1}\right)}{-e^{-1} + 2 - e^{-1}} = 1. \tag{9.18}$$

Returning to the general linear boundary value problem, we have shown that it is possible to form an equivalent initial value problem and seek s^* such that the solution of the initial value problem satisfies $y(b, s^*) = \beta$. We have shown that it is possible to write the initial value problem as a system of first-order differential equations. This then allows us to apply numerical initial value problem solvers.

The next approach we use to solve the boundary value problem is based upon superposition. The starting point is to consider the following problems:

$$H1 \equiv \begin{matrix} y_1' = v_1 \\ v_1' = -pv_1 - qy_1 \end{matrix}, \quad \begin{matrix} y_1(a) = 1 \\ v_1(a) = y_1'(a) = 0 \end{matrix}, \tag{9.19}$$

$$H2 \equiv \begin{matrix} y_2' = v_2 \\ v_2' = -pv_2 - qy_2 \end{matrix}, \quad \begin{matrix} y_2(a) = 0 \\ v_2(a) = y_2'(a) = 1 \end{matrix}, \tag{9.20}$$

$$I \equiv \begin{matrix} y_p' = v_p \\ v_p' = r - pv_p - qy_p \end{matrix}, \quad \begin{matrix} y_p(a) = \alpha \\ v_p(a) = y_p'(a) = 0 \end{matrix}. \tag{9.21}$$

By applying the superposition of the solutions to the three systems of differential equations above, we have that the solution of the initial value problem is

$$y = c_1 y_1 + c_2 y_2 + c_p y_p. \tag{9.22}$$

It can easily be shown that the solution in (9.22) satisfies the general value problem. Now we consider the initial conditions for the three systems to find the values for the three constants; c_1, c_2 and c_p. Therefore, considering $y(a)$, we have

$$\alpha = y(a) = c_1 \cdot 1 + c_2 \cdot 0 + \alpha \Rightarrow c_1 = 0.$$

Thus we do not need the first system for the solution y_1.

Next we consider the $s = v(a) = y'(a)$, which implies

$$s = v(a) = y'(a) = 0 \cdot 0 + c_2 \cdot 1 + 0 \Rightarrow c_2 = s.$$

Finally, considering the second boundary condition $\beta = y(b)$, we have

$$\beta = y(b) = 0 \cdot y_1(b) + c_2 \cdot y_2(b) + y_p(b) \Rightarrow c_2 \frac{\beta - y_p(b)}{y_2(b)}.$$

Combining all the information above, we can form the solution to the boundary value problem as

$$y = \left(\frac{\beta - y_p(b)}{y_2(b)} \right) y_2(x) + y_p(x). \tag{9.23}$$

Thus it is possible to solve the boundary value problem by numerically solving $H2$ and I. A feature to note here about the method above is that $s^* = c_2 = \frac{\beta - y_p(b)}{y_2(b)}$.

Another remark to consider about this derivation is that in practice we are only using the numerical solutions, and as such

$$\frac{\beta - \hat{y}_p(b)}{\hat{y}_2(b)} = \hat{s}^*,$$

where $\hat{\cdot}$ represents the numerical solution. This same situation is also true for the linear interpolation-based shooting method for the formula for s^* where

$$s^* + \frac{(s_2 - 2 = s_2)(\beta - y(b, s_1))}{y(b, s_2) - y(b, s_1)}$$

becomes

$$s_1 + \frac{(s_2 - 2_1)\left(\beta - y_N^{s_1}\right)}{y_N^{s_2} - y_N^{s_1}} = s^*,$$

where $y_i^{s_1} \simeq y(ih, s_1) \simeq y(a + ih, s_1)$, and $b - a = Nh$, h is the time and or space step. It is possible to show that $s* \to s^*$ as $h \to 0$ if the numerical scheme is consistent, as defined in the last chapter.

9.2.1 Nonlinear Problems

If we consider the nonlinear boundary value problem

$$y'' = yy', \quad x \in [0, 1], \quad \begin{matrix} y(0) = 1 \\ y(1) = 0 \end{matrix}, \tag{9.24}$$

then the associated system of first-order differential equations-based initial value problem for (9.24) is

$$\begin{matrix} y' = v \\ v' = vy \end{matrix}, \quad \begin{matrix} y(0) = 1 \\ v(0) = y'(0) = s \end{matrix}. \tag{9.25}$$

Now, let the solutions of the initial value problem in (9.25) be denoted by $y(x, s)$ and $v(x, s)$, along with $g(s) = y(b, s) - \beta$, we would like s^* to be such that $g(s^*) = y(b, s^*) - \beta = 0$.

To find the value for s^*, there are multiple techniques that we could apply: bisection, linear interpolation, and Newton-Raphson; it is the latter that we consider here. The general expression for the Newton-Raphson method is

$$s_{n+1} = s_n - \frac{g(s_n)}{g'(s_n)}, \tag{9.26}$$

where the subscript n refers to the nth iteration. The next feature to define is $g'(s_n)$. From the notation we have been using so far, we note that the prime indicates a derivative of some sorts, but for the problem we are considering here, we have

$$g'(s) \equiv \left.\frac{\partial y(x,s)}{\partial s}\right|_{x=b}. \tag{9.27}$$

Now let $u = \dfrac{\partial y}{\partial s}$; this then leads to $g'(s) = u(b)$. If we now consider the x derivative of u, we have

$$u' = \frac{du}{dx} \equiv \frac{d}{dx}\left(\frac{\partial y}{\partial s}\right) = \frac{\partial}{\partial s}\left(\frac{dy}{dx}\right) \equiv \frac{\partial v}{\partial s}. \tag{9.28}$$

Now let $w = \dfrac{\partial v}{\partial s}$; this leads to

$$\begin{aligned} w' = \frac{dw}{dx} &\equiv \frac{d}{dx}\left(\frac{\partial v}{\partial s}\right) = \frac{\partial}{\partial s}\left(\frac{dv}{dx}\right) = \frac{\partial}{\partial s}(yv), \\ &= \frac{\partial y}{\partial s}v + \frac{\partial v}{\partial s}y = uv + wy. \end{aligned} \tag{9.29}$$

Therefore, given the expressions in (9.28) and (9.29), it is possible to write the nonlinear boundary value problem as a linear system of differential equations in terms of u and w as

$$\begin{aligned} u' &= w, \\ w' &= uw + yw, \end{aligned} \tag{9.30}$$

where it is possible to integrate (9.30) forward to obtain $u(b) = g'(s)$ and be able to apply the Newton-Raphson solver.

It may be case that instead of the nonlinear boundary value problem defined in (9.24), we have the situation where the initial conditions are in terms of sum of different initial conditions for both the true solution and its first derivative as

$$y'' = f\left(x, y, y'\right), \quad x \in [0,1], \quad \begin{matrix} y(0) + y'(0) = 1 \\ y(1) = \beta \end{matrix}. \tag{9.31}$$

The associated system of initial value differential equations would be

$$\begin{aligned} y' &= v, \\ v' &= f(x, y, v), \end{aligned} \tag{9.32}$$

but now the initial conditions are $y'(0) = s$ and $y(0) = 1 - s$. The solution $y(x,s)$ of the initial value problem is sought so that with $s = s^*$, we have $y(b, s^*) = \beta$, or equivalently $y(0) = s_1$ and $y'(0) = s_2$,

which would have the constraint $s_1^* + s_2^* = 1$. Thus, we would have $y(x, s_1, s_2)$, and then seek s_i^*, such that $y\left(b, s_1^*, s_2^*\right) = \beta$. This approach can be quite difficult and messy to solve for.

Another situation that could occur for the nonlinear boundary value problem is

$$y'' = f\left(x, y, y'\right), \quad x \in [a, b], \qquad \begin{array}{l} y'(a) = \alpha, \\ y(b) = \beta, \end{array}$$

which would have the associated initial value problem of the form

$$\begin{array}{ll} y' = v, & y(a) = s, \\ v' = f(x, y, v), & v(a) = y'(a) = \alpha. \end{array}$$

It could also be the case that the nonlinear boundary value problem may be of the form

$$y'' = f\left(x, y, y'\right), \quad x \in [a, b], \qquad \begin{array}{l} y(a) = \alpha, \\ y'(b) = \beta, \end{array}$$

which leads to the associated linear system of initial value differential equations;

$$\begin{array}{ll} y' = v, & y(a) = \alpha, \\ v' = f(x, y, v), & v(a) = y'(a) = s. \end{array}$$

Exercise 9.2. *Consider the boundary value problem defined as*

$$y'' + y' = 1, \qquad \begin{array}{l} y(0) = 1, \\ y(1) = 1. \end{array} \tag{9.33}$$

1. *Determine the exact solution of* (9.33).
2. *Use the result from the theory of superposition presented above to determine the value of* s^* *for which the solution of the initial value problem*

$$y'' + y' = 1, \qquad \begin{array}{l} y(0) = 0, \\ y'(0) = s^*, \end{array} \tag{9.34}$$

   coincides with the solution of the boundary value problem in (9.33).
3. *Verify that the solution determine in part 1 above has the property* $y'(0) = s^*$, *the value determined in part 2.*

We now move on to one of the more commonly used numerical-based techniques for solving boundary value problems of *finite differencing*.

9.3 Finite Difference Methods

We first consider the set of **standard finite difference methods**, that can be applied to numerically solve the general second-order boundary value problem, which is defined as

$$L[y] \equiv y'' + p(x)\, y' + q(x)\, y = r(x), \quad x \in [0, 1], \qquad \begin{array}{l} y'(0) = \alpha, \\ y'(1) = \beta. \end{array} \tag{9.35}$$

However, the more general mixed boundary conditions could be of the form

$$\begin{aligned} y(0) + \gamma y'(0) &= \alpha, \\ y(1) + \delta y'(1) &= \beta. \end{aligned} \tag{9.36}$$

The situation where $\gamma = \delta = 0$ would be the Dirichlet boundary conditions.

The first numerical scheme we shall consider is the familiar **Euler method** from the initial value problems chapter; only this time we are just applying the scheme in space, and not time. The first-order derivative of y can be approximated as

$$\frac{y_{i+1} - y_i}{h} = f(x, y_i) \Rightarrow y' = f(x, y) \approx \frac{y(x_i + h) - y(x_i)}{h} - f(x_i, y(x_i)) = 0, \tag{9.37}$$

which can easily be rearranged to obtain

$$y_{i+1} = y_i + hf(x_i, y_i), \tag{9.38}$$

for $i = 0, 1, \ldots, N$, where the interval $[0, 1]$ has been divided into N equal spaces of length h, so $Nh = 1$.

We now consider a central difference approximation to $y'(x)$ as

$$y'(x) \approx \frac{y(x + h) - y(x - h)}{2h}. \tag{9.39}$$

An important property to note here about this approximation is that

$$\begin{aligned} \frac{y(x+h) - y(x-h)}{2h} &= \frac{\left(y(x) + hy'(x) + \frac{h^2}{2}y''(x) + \frac{h^3}{3!}y'''(\xi)\right) - \left(y(x) - hy'(x) + \frac{h^2}{2}y''(x) - \frac{h^3}{3!}y'''(\eta)\right)}{2h}, \\ &= y'(x) + \frac{h^2}{12}y'''(\xi) + \frac{h^2}{12}y'''(\eta), \\ &= y'(x) + \frac{h^2}{6}y'''(\zeta), \end{aligned}$$

where $\xi \in (x, x + h)$, $\eta \in (x - h, x)$, and $\zeta \in (x - h, x + h)$. Thus the central difference approach above is second order in space.

The central difference approximation for the second-order spatial derivative over the same interval as for the first-order derivative, $[x - h, x + h]$, is obtained through considering the difference between two different Euler methods for the first-order derivative as

$$y''(x) \approx \frac{\frac{y(x+h) - y(x)}{h} - \frac{y(x) - y(x-h)}{h}}{h} \equiv \frac{y(x + h) - 2y(x) + y(x - h)}{h^2}. \tag{9.40}$$

If we now expand the expression in (9.40) as a Taylor series about x, then we obtain

$$\begin{aligned} \frac{y(x + h) - 2y(x) + y(x - h)}{h^2} &= y''(x) + \frac{h^2}{4!}y''''(\xi) + \frac{h^2}{4!}y''''(\eta), \\ &= y'' + \frac{h^2}{12}y''''(\zeta), \end{aligned}$$

where again $\xi \in (x, x+h)$, $\eta \in (x-h, x)$, and $\zeta \in (x-h, x+h)$.

To simplify the expressions for the numerical schemes we introduce the following notation: $y(x_i) = y_i$ where $x_i = ih$, for $i = 0, 1, \ldots, N$ and $x_{i\pm1} \equiv x_i \pm h$, and $y_{i\pm1} \simeq y(x_i \pm h)$.

Substituting the two central differences approximation derived above into the general boundary value problem in (9.35), we obtain

$$L_h[y_i] = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i = r_i, \tag{9.41}$$

where $p_i = p(x_i)$, $q_i = q(x_i)$, and $r_i = r(x_i)$ and are the exact values for these functions. If we now consider the Dirichlet problem with the conditions $y(0) = \alpha$ and $y(1) = \beta$, then we would take $y_0 = \alpha$ and $y_N = \beta$. Thus, multiplying (9.41) by h^2, we obtain

$$\left(-1 - \frac{h}{2}p_i\right) y_{i+1} + \left(2 + h^2 q_i\right) y_i + \left(-1 + \frac{h}{2}p_i\right) y_{i-1} = h^2 r_i. \tag{9.42}$$

This holds for $i = 1, 2, \ldots, N-1$ where we have $y_0 = \alpha$ and $y_N = \beta$. Therefore, the difference equation in (9.42) results in system of $N-1$ equations for $N-1$ unknowns $y_1, y_2, \ldots, y_N$, which can be written in matrix-vector form $\mathbf{A}\boldsymbol{y} = \boldsymbol{r}$, where the matrix $\mathbf{A}$ is a tri-diagonal matrix. We now denote

$$a_i = -1 - \frac{h}{2}p_i, \quad b_i = 2 + h^2 q_i, \quad c_i = -1 + \frac{h}{2}p_i.$$

Given the definition for the a_is, b_is and c_is above, it is possible to write the numerical approximation as a matrix-vector equation given by

$$\begin{pmatrix} b_1 & c_1 & 0 & 0 & \cdots & \cdots & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \cdots & \vdots \\ \vdots & 0 & a_i & b_i & c_i & \ddots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \vdots & \cdots & \cdots & a_{N-2} & b_{N-2} & c_{N-2} \\ 0 & \cdots & \cdots & \cdots & \cdots & a_{N-1} & b_{N-1} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} -a_1\alpha + h^2 r_1 \\ h^2 r_2 \\ \vdots \\ h^2 r_i \\ \vdots \\ h^2 r_{N-2} \\ -c_{N-1}\beta + h^2 r_{N-1} \end{pmatrix}. \tag{9.43}$$

Matrix-vector equations of the form in (9.43) are easy to solve.

We now consider some properties of the matrix $\mathbf{A}$: If $\mathbf{A}$ is strictly diagonally dominant then $\mathbf{A}^{-1}$ exists. However, we have to consider when would this property occur. For the tri-diagonal matrix in (9.43) to be strictly diagonally dominant, we require

$$|b_i| > |a_i + c_i| > |a_i| + |c_i| \Rightarrow \left|2 + h^2 q_i\right| > \left|-1 - \frac{h}{2}p_i\right| + \left|-1 + \frac{h}{2}p_i\right|.$$

If we consider the diagonal entry b_i, then we have

$$|b_i| = \left|2 + h^2 q_i\right| \geq 2 \equiv 1 + \frac{h}{2}p_i + 1 - \frac{h}{2}p_i,$$

$$= -\left(-1 - \frac{h}{2}p_i\right) - \left(-1 + \frac{h}{2}p_i\right),$$
$$= -a_i - c_i = |a_i| + |c_i|,$$

provided that a_i and c_i are less than or equal to zero. Thus

$$-1 - \frac{h}{2}p_i \leq 0 \quad -1 + \frac{h}{2}p_i \leq 0.$$

Given the inequality above, it is possible to define bounds on the step size h through

$$-\frac{2}{p_i} \leq h \leq \frac{2}{p_i} \Rightarrow h \leq \frac{2}{|p_i|}, \quad \forall i. \tag{9.44}$$

Therefore, if $q(x) \geq 0$ and $h \leq \frac{2}{\max_{x\in[0,1]}|p(x)|} = \frac{2}{P}$, then we have $|b_i| \geq |a_i| + |c_i|$, with strict diagonal dominance $|b_i| > |a_i| + |c_i|$ if $q(x) > 0$. Therefore, if $q > 0$ and $h \leq \frac{2}{P}$ then $\mathbf{A}$ is strictly diagonally dominant, which implies that the inverse of $\mathbf{A}$, $\mathbf{A}^{-1}$, exists. Thus, there is a unique solution for $\boldsymbol{y}$.

We now consider the following example to help illustrate the properties that have been derived above. We are going to discretize the following boundary value problem,

$$-y'' + \pi^2 \sin(\pi x)\, y = 0, \quad x \in [0,1], \qquad \begin{matrix} y(0) = 0, \\ y(1) = 1, \end{matrix} \tag{9.45}$$

to determine if the resulting matrix satisfy the properties for a unique solution given above.

We start by forming the numerical mesh, where we have N intervals of length h, such that $Nh = 1$, this implies that $x_i = ih$ and $y(x_i) \simeq y_i$. Given these attributes for the numerical grid, the associated difference equation is

$$L_h[y_i] = -\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + \pi^2 \sin(\pi x_i)\, y_i = 0. \tag{9.46}$$

Rearranging (9.46) to be able to form the matrix-vector equation, we have

$$-y_{i-1} + \left(2 + \pi^2 \sin(\pi i h)\right) y_i - y_{i+1} = 0, \quad i = 1, 2, \ldots, N-1, \;\; y_0 = 0, \;\; y_N = 1.$$

Therefore, if we let $a_i = -1$, $b_i = 2 + h^2\pi^2 \sin(\pi i h)$, and $c_i = -1$, then the associated tri-diagonal matrix equation for this example is

$$\begin{pmatrix} b_1 & c_1 & 0 & \cdots & \cdots & \cdots & 0 \\ a_2 & b_2 & c_2 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \vdots & \ddots & a_i & b_i & c_i & 0 & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & a_{N_2} & b_{N-2} & c_{N-2} \\ 0 & 0 & 0 & \cdots & \cdots & a_{N-1} & b_{N-1} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}.$$

First, considering the bound on the step size h we notice that $p(x) \equiv 0$, therefore $h < \frac{2}{P} = \infty$. **Thus we can take any value for the step size**. Next we check whether or not there are values for b_i and a_i and c_i, so that the matrix is not strictly diagonal dominant, where we require $|b_i| < |a_i| + |c_i|$. Thus we have

$$|b_i| = \left|2 + h^2\pi^2 \sin(\pi i h)\right| = 2 + h^2\pi^2 \sin(\pi i h) > 2 = 1 + 1 = |-1| + |-1| = |a_i| + |c_i|,$$

and the matrix is diagonally dominant, and as such it is invertible and a unique solution exists to the numerical approximation. We now more on to consider the truncation errors associated with different numerical schemes.

9.3.1 Truncation Error

We now consider the truncation errors associated with different numerical schemes, and so we start with a definition of these errors, given by

$$\begin{aligned}\tau_i &= L_h[y(x)] - r_i,\\ &= L_h[y(x_i)] - L_h[y_i],\\ &= L_h[y(x_i) - y_i],\end{aligned} \tag{9.47}$$

where the expression inside the operator in (9.47) is the error in the solution defined as $e_i = y(x_i) - y_i$.

Returning to the central difference-based approximation to the second-order boundary value problem, we have

$$\begin{aligned}\tau_i &= L_h[y(x_i)] - r_i,\\ &= -\left(\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2}\right) + p(x_i)\left(\frac{y(x_{i+1}) - y(x_{i-1})}{2h}\right) + q(x_i)y_i - r_i.\end{aligned} \tag{9.48}$$

Expanding $x_{i\pm1} = x_i \pm h$ as a Taylor series and substituting into (9.48), we obtain

$$\begin{aligned}\tau_i = -\frac{1}{h^2}\Bigg(&y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \frac{h^3}{3!}y'''(x_i) + \frac{h^4}{4!}y''''(x_i) + \cdots - 2y(x_i) + y(x_i)\\ &- hy'(x_i) + \frac{h^2}{2}y''(x_i) - \frac{h^3}{3!}y'''(x_i) + \frac{h^4}{4!}y''''(x_i) - \cdots\Bigg)\\ &+ p(x_i)\frac{1}{2h}\Bigg(y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \frac{h^3}{3!}y'''(x_i) + \frac{h^4}{4!}y''''(x_i) + \cdots - y(x_i)\\ &+ hy'(x_i) - \frac{h^2}{2!}y''(x_i) + \frac{h^3}{3!}y'''(x_i) - \frac{h^4}{4!}y''''(x_i) + \cdots\Bigg) + q(x_i)y(x_i) - r(x_i).\end{aligned} \tag{9.49}$$

Many of the terms cancel in (9.49) through the opposite sign and as a result of the partial differential equation, which leaves

$$\tau_i = -\frac{h^2}{12}y''''(x_i) + p(x_i)\frac{h^2}{6}y'''(x_i) + \mathcal{O}\left(h^3\right),$$

$$= -\frac{h^2}{12} y''''(\xi_1) + p(x_i)\frac{h^2}{6} y'''(\xi_2), \quad \xi_1, \xi_2 \in (x_i - h, x_i + h),$$
$$= -\frac{h^2}{12} y''''(x_i + \theta h) + p(x_i)\frac{h^2}{6} y'''(x_i + \psi h), \quad \theta, \eta \in (-1, 1),$$
$$\Rightarrow |\tau_i| \leq \frac{h^2}{12}(M_4 + 2M_3), \tag{9.50}$$

where

$$M_4 = \max\left|y''''(x)\right|, \quad M_3 \equiv \left|y'''(x)\right|, \quad P \equiv \max|p(x)|, \quad x \in (0, 1).$$

Therefore, the difference scheme that we have presented here is a second-order method, where the truncation error will tend to zero at the rate of $h^2 \to 0$.

9.3.2 Mixed Boundary Conditions

In this situation we consider the case where the values of the solution at one or both boundaries is not given and so the numerical point at the boundary becomes an unknown in the problem. We therefore need to supplement the existing equations with an extra equation.

If we consider the example

$$-y'' + q(x)y = f(x), \quad x \in [0, 1], \qquad \begin{array}{l} y(1) = \beta, \\ y(0) - y'(0) = 0, \end{array} \tag{9.51}$$

then we can see from (9.51) that we do not have an explicit value for $y(0)$, and therefore we do not have a value for y_0 in our numerical approximation.

Applying the second-order approximation that we have already introduced results in

$$L_h[y_i] = -\frac{(y_{i+1} - 2y_i + y_{i-1})}{h^2} + q_i y_i = f_i,$$

and this can be rearranged to

$$-y_{i+1} + \left(2 + h^2 q_i\right) y_i - y_{i-1} = h^2 f_i, \quad i = 1, 2, \ldots, N - 1. \tag{9.52}$$

Thus the formulation above gives us $N - 1$ equations for N unknowns; we know that $y_N = \beta$ but we do have an expression for y_0 which is our extra unknown in this situation.

To overcome this shortfall we shall introduce a **fictitious point**, which is often referred to as a **ghost point**, and denote this point as y_{-1}. Since we are only short of one equation, we consider difference equation in (9.52) for $i = 0$ also which is

$$-y_{-1} + \left(2 + h^2 q_0\right) y_0 - y_1 = h^2 f_0. \tag{9.53}$$

To find an expression for y_{-1}, we consider a central difference approximation to the boundary condition at $x = 0$, which results in

$$y_0 - \frac{(y_1 - y_{-1})}{2h} = 0. \tag{9.54}$$

Rearranging (9.54) to eliminate y_{-1}, we obtain

$$y_{-1} = y_1 - 2hy_0. \tag{9.55}$$

Substituting (9.55) into (9.53) results in

$$\left(2 + 2h + h^2 q_0\right) y_0 - 2y_1 = h^2 f_0. \tag{9.56}$$

Combining (9.52) with (9.56) now gives us N equations for the N unknowns $y_0, y_1, \ldots, y_{N-1}$ and the matrix equation remains tri-diagonal.

Given this new equation for the initial or left boundary, we have to consider the impact on the truncation error that our approximation for y_{-1} has. For all of the equations we have $L_h[y_i] = f_i$ for $i = -1, 0, 1, \ldots, N-1$ and as such we have that the truncation error is $\tau_i = -\frac{h^2}{12} y''''(\eta)$ for $\eta \in (x_i - h, x_i + h)$. If we now consider the equation for y_0, then we have $L_h[y_0] = f_0$. Therefore, the truncation error for the first equation is

$$\begin{aligned} \tau_0 &= L_h[y(x_0)] - f_0, \\ &= L_h[y(0)] - f_0, \\ &= \frac{\left(2 + h^2 q(0) + 2h\right) y(0) - 2y(h)}{h^2} - f_0. \end{aligned} \tag{9.57}$$

The next step in the derivation of the truncation error at $x = 0$ is to expand $y(h)$ by noticing that this is equivalent to $y(0+h)$ and expanding as a Taylor series about $x = 0$. This results in

$$\tau_0 = \frac{1}{h^2}\left(\left(2 + h^2 q(0) + 2h\right) y(0) - 2\left(y(0) + hy'(0) + \frac{h^2}{2} y''(0) + \frac{h^3}{3!} y'''(0) + \cdots\right)\right) - f_0. \tag{9.58}$$

There are many cancelations that occur in (9.58): the first is due to the boundary condition at $x = 0$ which is $h\left(y(0) - y'(0)\right) = 0$ and the second is the partial differential equation itself at $x = 0$, that is to say, $-y''(0) + q(0) y(0) - f(0)$. This then leaves the truncation error as

$$\begin{aligned} \tau_0 &= -\frac{h}{3} y'''(0) + \cdots, \\ &= \mathcal{O}(h), \\ &= -\frac{h}{3} y'''(\eta), \quad \eta \in (o, h). \end{aligned}$$

Therefore, this approximation is not second order in terms of h, it is only $\mathcal{O}(h)$.

Exercise 9.3. *Discretize the boundary value problem*

$$-y'' + e^x y' + \sin(x)\, y = 0, \quad x \in [0, \pi],$$

for the two sets of boundary conditions

(i) $y(0) = 0,\ y(\pi) = 1$; *and*
(ii) $y(0) = 0,\ y(\pi) + y'(\pi) = 1.$

Write down the first, last, and the general equations for the two sets of boundary conditions above and determine that the truncation error τ_i for $i = 1, 2, \ldots, N-1$ for both sets of boundary conditions satisfy

$$|\tau_i| \leq \frac{h^2}{12}\left(M_4 + 2e^{\pi} M_3\right), \quad M_{3,4} \equiv \max_{x\in[0,\pi]} \left|y^{(3,4)}(x)\right|.$$

Determine a bound of τ_N for boundary conditions in (ii). Show for both sets of boundary conditions that the restriction on the grid spacing of $h \leq 2e^{-\pi}$ guarantees that the system of equations has a unique solution.

9.4 Self-Adjoint Problems

We now move on to consider a special class of boundary values problems, that are referred to as self-adjoint. A self-adjoint problems is of the form

$$-\frac{d}{dx}\left(py'\right) + q(x)\, y = r(x), \quad x \in [0,1], \quad \begin{array}{l} y(0) = \alpha, \\ y(1) = \beta, \end{array} \tag{9.59}$$

where $p(x) \geq 0$ and it could also be the case that we have mixed boundary conditions. Expanding the differential operator in the first term in (9.59), and dividing by p, we can then write (9.59) as

$$-y'' - p'y' + \frac{qy}{p} = \frac{r}{p}, \tag{9.60}$$

where we must have the property $p(x) \neq 0$ for $x \in [0,1]$. While it is possible to consider the differential equation in the form in (9.60), it would be good to utilize the property $\frac{d}{dx}\left(py'\right)$ with respect to applying integration.

To solve the differential equation in (9.59), we are going to introduce numerical integration-based methods. We have been using the notation that the numerical value for the solution at point $x_{i\pm1} \equiv x_i \pm h$. Now we introduce an additional point, halfway between x_i and $x_{i\pm1}$, as $x_{i\pm\frac{1}{2}} \equiv x_i \pm \frac{h}{2}$.

Therefore, we integrate (9.59) to remove the differential operator as

$$-\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \frac{d}{dx}\left(py'\right)dx + \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} qy\,dx = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} r\,dx. \tag{9.61}$$

The first term in (9.61) can be integrated exactly while the other two terms are approximated by $hq_i y_i$ and hr_i, respectively. Returning to the first term, we have

$$-\left(py'\right)\Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = -\left(p_{i+\frac{1}{2}} y'\left(x_i + \frac{h}{2}\right) - p_{i-\frac{1}{2}} y'\left(x_i - \frac{h}{2}\right)\right). \tag{9.62}$$

We now approximate the derivatives in (9.62) with an upwind and a downwind numerical scheme, which results in

$$-\left(py'\right)\Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = -\left(p_{i+\frac{1}{2}}\left(\frac{y_{i+1} - y_i}{h}\right) - p_{i-\frac{1}{2}}\left(\frac{y_i - y_{i-1}}{h}\right)\right).$$

This leads to the equivalent discrete equation for (9.61) as

$$-\left(p_{i+\frac{1}{2}}y'\left(x_i+\frac{h}{2}\right)-p_{i-\frac{1}{2}}y'\left(x_i-\frac{h}{2}\right)\right)+hq_iy_i=hr_i. \tag{9.63}$$

A feature to note about (9.63) relates to the truncation error for this numerical scheme. We denote the integrated operator version as $\hat{L}_h[y_i]=r_i$, which is equivalent to the difference equation in (9.63) divided by h, which implies that the truncation error is $\hat{\tau}_i=\mathcal{O}\left(h^2\right)$.

We now consider how to form the matrix-vector equation for

$$-p_{i-\frac{1}{2}}y_{i+1}+\left(p_{i-\frac{1}{2}}+p_{i+\frac{1}{2}}+h^2q_i\right)y_i+p_{i+\frac{1}{2}}y_{i-1}=h^2r_i, \tag{9.64}$$

with $i=1,2,\ldots,N-1$. As we are considering a boundary value problem with Dirichlet boundary conditions, we have $y_0=\alpha$ and $y_N=\beta$. Thus, it is possible to write (9.64) in terms of the coefficients of the matrix as

$$\begin{aligned}
&\hat{a}_iy_{i+1}+\hat{b}_iy_i+c_iy_{i-1}=h^2r_i,\\
&\hat{a}_i=-p_{i-\frac{1}{2}},\\
&\hat{b}_i=p_{i+\frac{1}{2}}+p_{i-\frac{1}{2}}+h^2q_i,\\
&\hat{c}_i=-p_{i+\frac{1}{2}}.
\end{aligned}$$

Therefore, we still have a tri-diagonal matrix but now the associated matrix for the discretization is symmetric (i.e., $\hat{a}_i=-p_{i+1-\frac{1}{2}}=p_{i+\frac{1}{2}}=\hat{c}_i$). Next we have to consider if the matrix is diagonally dominant. We consider the modulus of $\hat{b}_i$ and need to show that it is greater than or equal to the modulus of the sum of the off diagonal entries, which for a tri-diagonal system implies $\left|\hat{a}_i+\hat{c}_i\right|$. Thus, we have

$$\left|\hat{b}_i\right|=\left|p_{i-\frac{1}{2}}+p_{i+\frac{1}{2}}+h^2q_i\right|=p_{i-\frac{1}{2}}+p_{i+\frac{1}{2}}+h^2q_i\geq p_{i-\frac{1}{2}}+p_{i+\frac{1}{2}}=-\hat{a}_i-\hat{c}_i=\left|\hat{a}_i\right|+\left|\hat{c}_i\right|. \tag{9.65}$$

For the system to be diagonally dominant we require $p,q\geq 0$, and for the diagonally dominance criteria to be independent of h; moreover, if $q>0$, then we have strict diagonal dominance.

We now consider the case where we have mixed boundary condition and assess the affect these have on the order of accuracy of the numerical scheme. Therefore, we are considering the boundary value problem as

$$-\left(py'\right)+qy=r,\quad x\in[0,1],\qquad \begin{aligned}&y(0)-\alpha y'(0)=1,\\&y(1)=\beta,\end{aligned} \tag{9.66}$$

where we assume that $\alpha\neq 0$. The associated discrete approximation for the interior of the domain is still

$$-p_{i+\frac{1}{2}}y_{i-1}+\left(p_{i+\frac{1}{2}}+p_{i-\frac{1}{2}}+h^2q_i\right)y_i+p_{i-\frac{1}{2}}y_{i-1}=h^2r_i,$$

for $i-1,2,\ldots,N-1$.

To evaluate the mixed boundary conditions, we start by integrating the self-adjoint boundary value problem in (9.59). We integrate the differential equation from x_0 to $x_{\frac{1}{2}}$, that is to say, from $x = 0$ to $x = \frac{h}{2}$, which is just to the right of the left boundary.

$$-\int_0^{\frac{h}{2}} \frac{d}{dx}\left(py'\right)dx + \int_0^{\frac{h}{2}} qydx = \int_0^{\frac{h}{2}} rdx. \tag{9.67}$$

Evaluating each term in (9.67) starting from the right, we have

$$\begin{aligned}
\int_0^{\frac{h}{2}} rdx &\approx \frac{hr_0}{2}, \\
\int_0^{\frac{h}{2}} qydx &\approx \frac{h}{2} q_0 y_0, \\
\int_0^{\frac{h}{2}} -\frac{d}{dx}\left(py'\right)dx &= -py'\Big|_0^{\frac{h}{2}}, \\
&= -\left(p\left(\frac{h}{2}\right)y'\left(\frac{h}{2}\right) - p\left(0\right)y'\left(0\right)\right).
\end{aligned}$$

Applying the upwind to the derivative term in the first component above between y_0 and y_1 will have the derivative for $y_{\frac{1}{2}}$; we have $p_{\frac{1}{2}} \frac{y_1 - y_0}{h}$. For the second term we rearrange the boundary condition to replace $y'\left(0\right) \equiv \frac{y_0 - 1}{\alpha}$, which results in

$$\left(p\left(\frac{h}{2}\right)y'\left(\frac{h}{2}\right) - p\left(0\right)y'\left(0\right)\right) \approx -\left(p_{\frac{h}{2}} \frac{y_1 - y_0}{h}\right) - p_0\left(\frac{y_0 - 1}{\alpha}\right). \tag{9.68}$$

Factorizing (9.68) results in

$$\left(\frac{p_0}{\alpha}h + \frac{h}{2}q_0 + p_{\frac{h}{2}}\right)y_0 - p_{\frac{h}{2}} y_1 = \frac{h^2}{2}r_0 + \frac{p_0}{\alpha}h. \tag{9.69}$$

Combining (9.69) with (9.64) results in N equations in N unknowns, but more importantly we have been able to retain the symmetry of the matrix.

9.5 Error Analysis

We return to the non self-adjoint differential equation to determine the framework for the error analysis. We have already defined the operator, $L_h\left[y_i\right] = r_i$, where $h^2 L_h\left[y_i\right] = a_i y_{i-1} + b_i y_i + c_i y_{i+1}$ and the coefficients are

$$a_i = -\frac{h}{2}p_i, \quad b_i = 2 + h^2 q_i, \quad c_i = -1 + \frac{h}{2}p_i.$$

The error we consider here is defined as $e_i = y\left(x\right) - y_i$, which leads to $h^2 L_h\left[e_i\right] = h^2 \tau_i$. We know that $L_h\left[y_i\right] = r_i$ and we also have that the truncation error is defined, $L_h\left[y\left(x_i\right)\right] - r_i = \tau_i$, which leads to

$$L_h\left[e_i\right] = L_h\left[y\left(x_i\right)\right] = L_h\left[y_i\right] = \tau_i + r_i - r_i = \tau_i. \tag{9.70}$$

We know that we can express the operator L_h acting on e_i in the left-hand side of the equation in (9.70) in terms of the coefficients just defined, which results in

$$a_i e_{i-1} + b_i e_i + c_i e_{i+1} = h^2 \tau_i. \tag{9.71}$$

From the earlier work, we already know that

$$|b_i| \geq |a_i| + |c_i| \quad \text{if } q_i \geq 0, \quad h \leq \frac{2}{p_i}.$$

Moreover, we know that if $b_i \geq 2$ and a_i and c_i are less than or equal to zero then $b_i \geq -a_i - c_i$, which implies that $\left|2 + h^2 q_i\right| \geq \left|-1 - \frac{h}{2} p_i\right| + \left|-1 + \frac{h}{2} p_i\right|$. We now consider different situations that could occur for the functions $q(x)$ and $p(x)$.

Case 1: Easy Case. For the easy case we are going to consider the situation where $q(x) \geq Q > 0$. Therefore, from (9.71) we have

$$\begin{aligned}
&\left(-1 - \frac{h}{2} p_i\right) e_{i-1} + \left(2 + h^2 q_i\right) e_i + \left(-1 + \frac{h}{2} p_i\right) e_{i+1} = h^2 \tau_i, \\
&\Rightarrow \left(2 + h^2 q_i\right) e_i = \left(-1 - \frac{h}{2} p_i\right) e_{i-1} + \left(1 + \frac{h}{2} p_i\right) e_{i-1} + \left(1 - \frac{h}{2} p_i\right) e_{i+1} + h^2 \tau_i, \\
&\Rightarrow \left|\left(2 + h^2 q_i\right) e_i\right| = \left|\left(-1 - \frac{h}{2} p_i\right) e_{i-1} + \left(1 + \frac{h}{2} p_i\right) e_{i-1} + \left(-\frac{h}{2} p_i + 1\right) e_{i+1} + h^2 \tau_i\right|, \\
&\Rightarrow \left(2 + h^2 Q\right) |e_i| \leq \left|\left(2 + h^2 q_i\right) e_i\right| \leq \left(1 + \frac{h}{2} p_i\right) |e_{i-1}| + \left(1 - \frac{h}{2} p_i\right) |e_{i+1}| + h^2 |\tau_i|.
\end{aligned} \tag{9.72}$$

We now suppose that the maximum of all the errors is denoted as $e_{\max}$, the maximum value for the truncation error as $\tau_{\max}$, which are defined as

$$\max_{1 \leq i \leq N-1} |e_i| = e_{\max}, \qquad \max_{1 \leq i \leq N-1} \tau_i = t_{\max}, \tag{9.73}$$

where $e_0 = e_N = 0$ due to the Dirichlet boundary conditions. Therefore, we have

$$\begin{aligned}
\left(2 + h^2 Q\right) |e_i| &\leq \left(1 + \frac{h}{2} p_i\right) e_{\max} + \left(1 - \frac{h}{2} p_i\right) e_{\max} + h^2 \tau_{\max}, \\
&= 2 e_{\max} + h^2 \tau_{\max}.
\end{aligned}$$

If $\max_{1 \leq i \leq N-1} |e_i| = |e_k|$ for some k, then we have

$$\begin{aligned}
\left(2 + h^2 Q\right) |e_k| &\leq 2 e_{\max} + \tau_{\max} h^2, \\
\left(2 + h^2 Q\right) e_{\max} &\leq 2 e_{\max} + \tau_{\max} h^2, \\
\Rightarrow e_{\max} &\leq \frac{\tau_{\max}}{Q}.
\end{aligned} \tag{9.74}$$

Substituting the expression for $\tau_{\max}$ results in the bound for the error associated with the central difference approximations to the general boundary condition problem as

$$\max_{1\le i\le N_1}|e_i| \le \frac{1}{Q}\left[\frac{h^2}{12}\left(\max_{x\in[0,1]} y''''(x) + 2P\max_{x\in[0,1]} y'''(x)\right)\right],$$
$$= \frac{h^2}{12Q}\left(M_4 + 2PM_3\right), \tag{9.75}$$

where we have $Q = \min_{x\in[0,1]} q(x) > 0$ and $P = \max_{x\in[0,1]}|p(x)|$. Therefore, we have that $e_i = y(x_i) - y_i \to 0$ as $h \to 0$, so $y_i \to y(x)$, and converges as $\mathcal{O}\left(h^2\right)$.

Case 2: Where $q(x) \ge 0$, $h \le \frac{2}{P}$, $q_i \ge 0$. To be able to determine the convergence of the central difference scheme for the boundary value problem when $q(x) \ge 0$, we require a maximum principle which is given by the following theorem.

Theorem 9.4. *The operator L_h defined by $h^2L_h\left[v_j\right] = a_jv_{j-1} + b_jv_j + c_jv_{j+1}$, where $b_j > 0$ and a_j and c_j are less than or equal to zero and $b_j \ge -a_j - c_j$ satisfies a* ***maximum principle****. That is to say that if $L_h\left[v_j\right] \le 0$, then $\max_{1\le j\le N-1} v_j \le \max\{v_0, v_N, 0\}$.*

This theorem is stating that the maximum of v_j occurs at a boundary, which means that v_j cannot attain a non-negative maximum at an interior point.

Proof. We had a maximum principle theorem in the initial value problem chapter, but that was with respect to discretization for the heat equation. Here we shall prove this theorem for the central difference scheme we have been proving different properties of so far in this chapter.

Before we start, we introduce a constant M such that $\max_{1\le j\le N-1} v_j = M$. As with the proof for the maximum principle for the discretization of the heat equation, we need to show that $M \le \max\{v_0, v_N, 0\}$. If we first assume that $M \le 0$, then we have that the left-hand side of the inequality is less than or equal to zero, and the right-hand side of the inequality is either greater than or equal to zero and therefore the inequality holds.

We now consider the remaining case of $M > 0$, then we have $M = v_k > 0$ for some k between 1 and $N-1$, otherwise the result would be proven as v_0 and v_N are part of the bounding inequality.

To prove that $v_k \le \max\{v_0, v_N, 0\}$ we form a proof by contradiction. Therefore, we assume that $v_k > \max\{v_0, v_N, 0\}$. Applying the discrete operator to v_j gives us

$$h^2L_h\left[v_j\right] = a_kv_{k-1} + b_kv_k + c_kv_{k-1} \le 0. \tag{9.76}$$

We also have $b_k \ge -a_k - c_k$ and $v_k > 0$, which implies that

$$b_kv_k \ge -a_kv_k - c_kv_k. \tag{9.77}$$

Combining (9.76) and (9.77) results in

$$-a_kv_k - c_kv_k \le b_kv_k \le -a_kv_{k-1} - c_kv_{k+1},$$

and thus

$$-a_k\left(v_{k-1} - v_k\right) - c_k\left(v_{k+1} - v_k\right) \ge 0, \tag{9.78}$$

with $-a_k \ge 0$ and $-c_k \ge 0$.

Now we have that $v_k = \max_{i\le j\le N-1} v_j \ge v_{k-1}, v_{k+1}$; therefore, $v_{k-1} - v_k \le 0$ and $v_{k+1} - v_k \le 0$. Thus the only way that (9.78) can be true is with $v_{k-1} - v_k = 0$ and $v_{k+1} - v_k = 0$. This then implies that $v_{k-1} = v_{k+1} = M$. Now either one of these is a boundary point, in which case this contradicts

the inequality, $v_k > \max\{v_0, v_N, 0\}$, or if not, apply same argument to $v_{k\pm1}$, so that $v_{k\pm2} = M$, and eventually the boundary is reached, which contradicts the inequality. Hence $M \leq \max\{v_0, v_N, 0\}$.

Returning to the formulation for case 2, we introduce

$$v_j = -\sigma g_j \pm e_j, \tag{9.79}$$

where $e_j \equiv y\left(x_j\right) - y_j$, $\sigma \equiv \max_{1\leq j\leq N-1}\left|\tau_j\right|$, $g_j \geq 0$ and $L_h\left[g_j\right] \geq 1$. The quantity $g_i \equiv g\left(x_i\right)$ is referred to as the **auxiliary function**, which will frequently depend on the problem that is being considered.

We now apply the discretization operator to (9.79), which results in

$$\begin{aligned} L_h\left[v_j\right] &= L_h\left[-\sigma g_i \pm e_i\right] = -\sigma L_h\left[g_i\right] \pm L_h\left[e_i\right], \\ &= -\sigma L_h\left[g_i\right] \pm \tau_i \leq -\sigma L_h\left[g_i\right] + \sigma = \sigma\left(1 - L_h\left[g_i\right]\right) \leq 0. \end{aligned} \tag{9.80}$$

Applying the maximum principle we have that $v_j \leq \max_{1\leq j\leq N-1} v_j \leq \max\{v_0, v_N, 0\}$, which implies

$$\begin{aligned} -\sigma g_i \pm e_i &\leq \max\{-\sigma g_0 \pm e_0, -\sigma g_N \pm e_N, 0\}, \\ &= \max\{-\sigma g_0, -\sigma g_N, 0\} = 0, \end{aligned} \tag{9.81}$$

where we have used the fact that we know the value of the function on the boundary and the associated errors at the boundaries, e_0 and e_N, are equal to zero. We also have $\pm e_j \leq \sigma g_j$, which implies that $\left|e_j\right| \leq \sigma\left|g_j\right| = \sigma g_j$. This then leads to

$$\max_{1\leq j\leq N-1}\left|e_j\right| \leq \sigma \max_{x\in[0,1]} g\left(x\right) = \sigma g_{\max} = \frac{h^2}{12}\left(M_4 + 2PM_3\right) g_{\max}. \tag{9.82}$$

To be able to evaluate the inequality above, we still need an auxiliary function, $g\left(x\right)$, where $g_i = g\left(x_i\right)$ has the property that $L_h\left[g_i\right] \geq 1$ for $g_i \geq 0$.

To help illustrate this principle, we consider the following example:

$$-y'' + q\left(x\right)y = r\left(x\right), \quad x \in [0,1], \qquad \begin{array}{l} y\left(0\right) = \alpha, \\ y\left(1\right) = \beta, \end{array} \tag{9.83}$$

for $q \geq 0$. We then have the question: What is the required auxiliary function? If we let $g\left(x\right) = \frac{1}{2}\left(x - x^2\right)$, then we have the property that $g\left(x\right) \geq 0$ for $x \in [0,1]$ and $L_h\left[g_i\right] \approx L\left[g\right] = -y''$. The maximum of $g\left(x\right)$ for $x \in [0,1]$ is $\frac{1}{8}$. This enables us to consider the maximum error in the interior as

$$1 \leq j \leq N-1 \leq \frac{h^2}{12}\frac{M_4}{8}. \tag{9.84}$$

Applying the discrete operator to the auxiliary functions results in

$$L_h\left[g_j\right] \approx L\left[y\right] = -y'' + qy = r\left(x\right).$$

We require $L_h\left[g_j\right] \geq 1$, therefore

$$
\begin{aligned}
L_h[g_j] &= -\frac{g_{i+1} - 2g_i + g_{i-1}}{h^2} + q_i g_i, \\
&= -\frac{1}{h^2}\left(\frac{1}{2}\left(x_{j+1} - x_{j+1}^2\right) - \left(x_j - x_j^2\right) + \frac{1}{2}\left(x_{j-1} - x_{j-1}^2\right)\right) + q_j \frac{1}{2}\left(x_j + x_j^2\right), \\
&= -\frac{1}{h^2}\left(\frac{x_j + h - x_j^2 - 2x_j h + h^2}{2} - x_j + x_j^2 + \frac{x_j - h - x_j^2 + 2x_j h - h^2}{2}\right) - q_j g_j, \\
&= 1 + q_j g_j \geq 1,
\end{aligned}
$$

where we have used the property that $g_{j+1} \equiv g_j + h$ and $g_{j-1} \equiv g_j - h$. The inequality above will hold if $q_j g_j \geq 0$. This then raises the question: When is there a unique solution to the matrix system? We know that if $h \leq \frac{2}{P_i}$ and $q_i > 0$ then the matrix, $\mathbf{A}$, is strictly diagonally dominant and therefore there exists an inverse for the matrix $\mathbf{A}$. However, if $q_i \geq 0$, then we cannot apply the results.

Before we continue, we need to introduce a definition for what is referred to as an M-matrix.

Definition 9.5. A matrix is called an M-matrix if the following three properties hold:

1. $a_{ii} > 0, a_{ij} < 0$.
2. $\mathbf{A}$ is diagonally dominant and is strictly diagonally dominant for at least one row.
3. The matrix $\mathbf{A}$ is irreducible.

We shall address what **irreducible** means after the following theorem.

Theorem 9.6. *If the matrix $\mathbf{A}$ is an M-matrix, then $\mathbf{A}$ is invertible.*

As an example, if we consider the numerical approximation to $-y'' = r$, then the associated matrix $\mathbf{A}$ for a central difference approximation to this differential equation is $\mathbf{A} = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}$, and the matrix $\mathbf{A}$ is an M-matrix and hence it is invertible.

9.5.1 Irreducibility

Before we can introduce the property of irreducibility of a matrix, we have to define the graph of a matrix.

Definition 9.7. The graph of the matrix $\mathbf{A}$ is a set of vertices, one for each row, and a directed line is drawn from vertex i to vertex j if $a_{ij} \neq 0$.

If we consider the example above where

$$
\mathbf{A} = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix},
$$

then we have $a_{ii} > 0$ for $i = 1, 2, 3$, and $a_{i,j} \leq -1$; therefore, the matrix $\mathbf{A}$ is diagonally dominant for one row and is strictly diagonally dominant for two rows. We have drawn the graph of the matrix in Fig. 9.3, where we can see that it is possible to travel from any vertex to any other vertex; therefore, the graph is referred to as being **strongly connected**, and then so is the matrix, $\mathbf{A}$. Therefore, the matrix $\mathbf{A}$ is an M-matrix, which implies that $\mathbf{A}$ is non-singular. Essentially this is the equivalent to saying that the numerical mesh is strongly connected. We now move on to consider numerical approximation to boundary value partial differential equations in at least two dimensions.

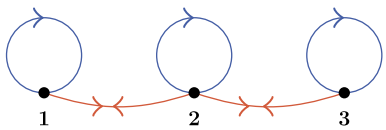


FIGURE 9.3

Graph of the matrix $\mathbf{A}$, to illustrate that it is strongly connected.

9.6 Partial Differential Equations

We start by defining the standard elliptical partial differential equations problem as

$$L[u] = -au_{xx} - bu_{yy} + cu_x + du_y + eu = f, \tag{9.85}$$

where a, b, c, d, e, f are functions of x and y and where the partial differential equation in (9.85) is for $u = u(x, y)$ on the domain $\mathcal{D}$ with $u = g(x, y)$ on the boundary $\mathcal{C}$ of $\mathcal{D}$. We have created an example of a domain and the associated boundary in Fig. 9.4.

The first step in numerically solving (9.85) is to introduce the mesh, or grid, that we shall use for a finite difference approach. We have taken the same domain from Fig. 9.4 and overlapped it with a numerical mesh in Fig. 9.5. Therefore, each point in the mesh in Fig. 9.5 is labeled as (x_i, y_j).

For the finite difference approach we shall have an approximate solution to the partial differential equation that is denoted as $u_{ij} \simeq u(x_i, y_j)$ at each grid point. The step sizes for the two spatial directions have been marked in Fig. 9.5, for the x-direction by h and for the y-direction by k. Another feature to note in Fig. 9.5 is that the mesh does not always coincide with the boundary $\mathcal{C}$, so we can only solve for u in the domain, where the unknowns are contained with this, and then incorporate the given boundary values in some way.

We begin with the partial differential equation

$$-\nabla^2 u \equiv -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} \equiv -u_{xx} - u_{yy} = f, \tag{9.86}$$

with $u = g$ and the boundary $\mathcal{C}$, where $\mathcal{C}$ is the boundary of $\mathcal{D}$ which for this problem is a rectangle.

The numerical solution is denoted by $u_{i,j} = u(x_i, y_j) = u(ih, jk)$. To approximate the partial differential equation in the rectangular domain, where $x \in (0, a)$ and $y \in (0, b)$, we shall use a central

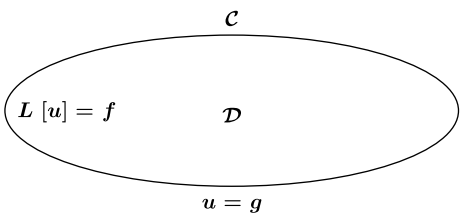


FIGURE 9.4

Diagram of a possible domain and its boundary for a general partial differential equation.

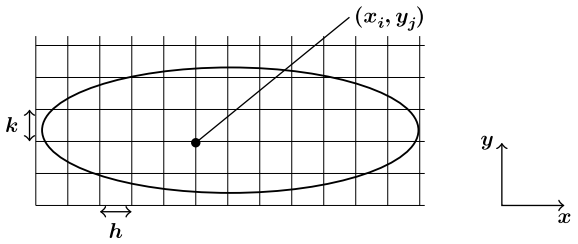


FIGURE 9.5

Diagram of a possible domain and its boundary with a numerical grid overlaid for a general partial differential equation.

difference, as we did for the ordinary differential equation for the x component, but here for both spatial directions, where the x-direction we have

$$\frac{\partial^2 u}{\partial x^2} \approx -\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}, \tag{9.87}$$

and the numerical approximation in the y-direction is

$$\frac{\partial^2 u}{\partial y^2} \approx -\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j+1}}{h^2}, \tag{9.88}$$

which makes the numerical approximation to (9.86)

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} \approx -\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} - \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j+1}}{h^2} = f_{i,j}, \tag{9.89}$$

where $f_{i,j} = f\left(x_i, y_j\right)$.

9.6.1 Truncation Error

The truncation error of the numerical scheme in (9.89) is defined as

$$\tau_{i,j} = -\nabla_h^2 u\left(x_i, y_j\right) - f_{i,j}, \tag{9.90}$$

where ∇_h^2 is the numerical approximation on the left-hand side of (9.89). Therefore, we have

$$\tau_{i,j} = -\frac{u\left(x_{i+1}, y_j\right) + 2u\left(x_i, y_j\right) + u\left(x_{i-1}, y_j\right)}{h^2} - \frac{u\left(x_i, y_{j+1}\right) + 2u\left(x_i, y_j\right) + u\left(x_i, y_{j-1}\right)}{k^2} - f_{i,j}. \tag{9.91}$$

The next step in the derivation of the truncation error is to expand (9.91) about $\left(x_i, y_j\right)$, where we have $x_{i\pm1} = x_i \pm h$ and $y_{j\pm1} = y_j \pm k$, which results in

$$\tau_{ij} = -\frac{1}{h^2}\left(u + hu_x + \frac{h^2}{2}u_{xx} + \frac{h^3}{3!}u_{xxx} + \frac{h^4}{4!}u_{xxxx}\left(\xi, y_j\right) - 2u + u - hu_x + \frac{h^2}{2}\right.$$

$$-\frac{h^3}{3!}u_{xxx}+\frac{h^4}{4!}u_{xxxx}(\eta,y_j)\Big)-\frac{1}{h^2}\Big(u+ku_y+\frac{k^2}{2}u_{yy}+\frac{k^3}{3!}u_{yyy}+\frac{k^4}{4!}u_{yyyy}(x_i,\gamma)$$
$$-2u+u-ku_y+\frac{k^2}{2}u_{yy}-\frac{k^3}{3!}u_{yyy}+\frac{k^4}{4!}u_{yyyy}(x_i,\delta)\Big)-f_{ij}, \tag{9.92}$$

where $\xi\in(x_i,x_i+h)$, $\eta\in(x_i-h,x_i)$ and $\gamma\in\big(y_j,y_j+k\big)$, $\delta\in\big(y_j-k,y_j\big)$.

After several terms cancel in (9.92), we are left with the following terms in the truncation error:

$$\tau_{i,j}=-u_{xx}-\frac{h^2}{4!}u_{xxxx}(\xi,y_j)-\frac{h^2}{4!}u_{yyyy}(x_i,\eta)-u_{yy}-\frac{h^2}{4!}u_{yyyy}(x_i,\gamma)-\frac{h^2}{4!}u_{yyyy}(x_i,\delta)-f_{i,j}. \tag{9.93}$$

The expression for the truncation error in (9.93) can be simplified further by noticing that it contains the partial differential equation that we are approximating. Therefore, the final expression for the truncation error for the central difference scheme for (9.85) is

$$\tau_{i,j}=-\frac{h^2}{12}\left(u_{xxxx}(\zeta,y_j)+u_{yyyy}(x_i,\varphi)\right), \tag{9.94}$$

where $\zeta\in(x_i-h,x_i+h)$ and $\varphi\in\big(y_j-k,y_j+k\big)$. Finally to bound the truncation error for this scheme, we have

$$|\tau_{i,j}|\le\frac{h^2}{12}\left(M_x^4+M_y^4\right), \tag{9.95}$$

where

$$M_{x(y)}^4=\max_{(x,y)\in\mathcal{D}}\left|u_{xxxx(yyyy)}(x_i,y_j)\right|.$$

We return to the numerical approximation and consider the case where the grid points are equidistant. Equidistant implies that $h=k$, which allows us to write the numerical scheme into a form similar to the matrix-vector equation we had for the numerical approximation to the ordinary differential equation as

$$4u_{i,j}-u_{i-1,j}-u_{i+1,j}-u_{i,j-1}-u_{i,j+1}=h^2f_{i,j}.$$

Another way to consider the approximation to $-\nabla^2u=f$ is to view the stencil for this approximation as the points of the compass, as shown in Fig. 9.6. This enables the approximation above to be expressed in a simpler form than with all the different subscripts, as

$$4u_p-u_E-u_W-u_S-u_N=h^2f_p.$$

This approximation is referred to as the **standard five-point central difference** approximation to $-\nabla^2u=f$ and the points in Fig. 9.6 are referred to as the **five-point stencil**.

We now consider solving the Laplace equation, recalling that this is the situation where $-\nabla^2u=f$ when $f=0$, on a rectangle defined by $[0,4]\times[0,3]$, with $u=g=1$ on the boundary of the rectangle.

To apply the numerical approximation to the Laplace equation, and then solve for the unknown values of u in the interior of the domain, we need a label for the unknowns and so we use a single

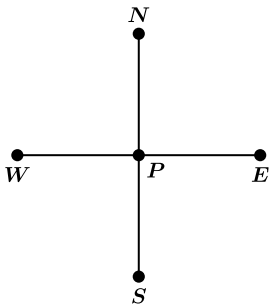


FIGURE 9.6

The five-point stencil for the central difference approximation to the Laplace or Poisson equation expressed as points of the compass.

index that runs from left to right on the x domain, and then upwards for the y domain. This ordering is referred to as **natural ordering**. Applying this description results in

$$\begin{aligned} 4u_1 - u_2 - u_4 &= 2, \\ 4u_2 - u_1 - u_3 - u_5 &= 1, \\ 4u_3 - u_2 - u_6 &= 2, \\ 4u_4 - u_1 - u_5 &= 2, \\ 4u_5 - u_4 - u_2 - u_6 &= 1, \\ 4u_6 - u_5 - u_3 &= 2. \end{aligned}$$

Forming the associated matrix-vector equation for the equations above results in

$$\left(\begin{array}{ccc|ccc} 4 & -1 & 0 & -1 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & -1 & 4 & 0 & 0 & -1 \\ \hline -1 & 0 & 0 & 4 & -1 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 \\ 0 & 0 & -1 & 0 & -1 & 4 \end{array}\right)\left(\begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{array}\right) = \left(\begin{array}{c} 2 \\ 1 \\ 2 \\ 2 \\ 1 \\ 2 \end{array}\right), \tag{9.96}$$

which is of the form $\mathbf{A}\boldsymbol{u} = \boldsymbol{f}$. We now consider the structure of the matrix in (9.96), which is of the form

$$\begin{array}{c|c} \mathbf{B}_1 & \mathbf{C}_1 \\ \hline \mathbf{A}_1 & \mathbf{B}_2 \end{array},$$

where the $\mathbf{B}_i$s are block matrices that are tri-diagonal of the form $\begin{matrix} -1 & 4 & -1 \\ -1 & 0 & 0 \end{matrix}$ and $\mathbf{A}_1 = \mathbf{C}_1$ are block diagonal matrices of the form $\begin{matrix} 0 & -1 & 0 \\ 0 & 0 & -1 \end{matrix}$. The ordering that for the mesh point labeling in the approximation is called the natural ordering. However, there are other orderings that we could have used to number the points in the interior of the domain. The first is referred to as the *red/black* ordering

or the *chequers board*. The next ordering is referred to as the *diagonal* ordering. A third alternative ordering is the *counter-clockwise* ordering. We have presented three different possible orderings for the mesh point labeling in Fig. 9.7.

9.6.2 General Natural Ordering

In the previous derivation for the numerical approximation for $\nabla^2 u = f$, we only had six points in the interior of the rectangular domain, this was due to the step sizes $h = k = 1$, but it is quite possible that the step size could have been a lot smaller and as such there would be more entries in the resulting $\mathbf{A}$ matrix between the points for the five-point stencil.

In Fig. 9.8 we produced a general mesh where there are I points in the x-direction and J points in the y-direction. Therefore, we have an $I \times J$ mesh, which is also the total number of points. We can see that there are many points in between the u_S point and when we arrive at the u_E point, and then again from the u_W point to the u_N point. This would imply that there is some interesting structure to the $\mathbf{A}$ matrix. The associated general row of the $\mathbf{A}$ matrix is given by

$$\begin{pmatrix} 0 & 0 & -1 & 0 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

It is possible to write the matrix $\mathbf{A}$ as

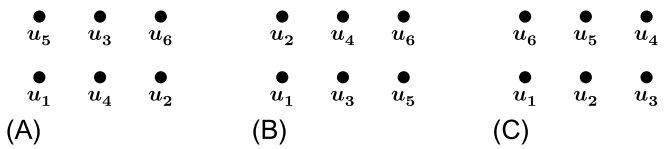


FIGURE 9.7

Examples of the (A) *red/black* (chequer board), (B) *diagonal*, and (C) *counter-clockwise* orderings.

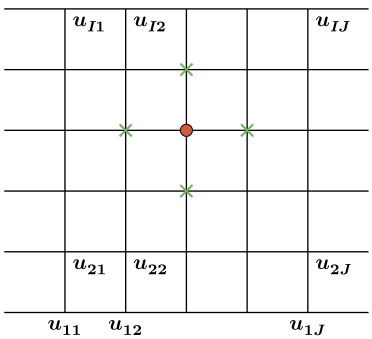


FIGURE 9.8

Numerical grid with natural ordering.

$$\mathbf{A} = \begin{pmatrix} \mathbf{B_1} & \mathbf{C_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{A_2} & \mathbf{B_2} & \mathbf{C_2} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A_3} & \mathbf{B_3} & \mathbf{C_3} & \mathbf{0} & \mathbf{0} \\ & & \ddots & \ddots & \ddots & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A_J} & \mathbf{B_J} \end{pmatrix}$$

The array above is referred to as a **block tri-diagonal** matrix, where the matrices $\mathbf{B_i}$, for $i = 1, 2, \ldots, J$ are themselves tri-diagonal, and the $\mathbf{A}$ and $\mathbf{C}$ matrices are diagonal matrices. We should note that $\mathbf{A_{i+1}} = \mathbf{C_i}$.

If we take for an example that we have a 50×50 mesh then we would have 2500 points, but for the five-point stencil we only require five points on each row. Therefore, there are at most 250 non-zero entries. This would make the associated matrix very a **sparse** matrix, but well structured. We would not use a direct method to invert this matrix but a technique designed for sparse systems, where these techniques for sparse matrix equations usually involve iterative schemes.

We now consider the uniqueness of the solution to the matrix equation above, where we have that $a_{i,i} = 4 > 0$ and $a_{i,j} < 0$. Therefore, the matrix is diagonally dominant and is strictly diagonally dominant for at least one row, where the latter property of $\mathbf{A}$ comes from the boundary conditions. We now require $\mathbf{A}$ to be irreducible. To assess this property, we have plotted the graph for the matrix, $\mathbf{A}$, in Fig. 9.9. The reason for plotting the graph is that we require the graph to be **strongly connected**.

The reason for considering whether the graph of the matrix is strongly connected is due to the property that we mentioned before, that if a graph is strongly connected then $\mathbf{A}$ is an $\mathbf{M}$-matrix, which implies that $\mathbf{A}$ is non-singular.

9.6.3 Error Bound on Numerical Solution

The continuous problem that we are trying to solve here is $-\nabla^2 u = f$ in the domain $\mathcal{D}$ where $u = g$ on the boundary, $\mathcal{C}$. The associated difference equation is $-\nabla_h^2 u_j - f_{i,j} = f_{i,j}$, which is the five-point stencil

$$\frac{1}{h^2}\left(4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1}\right) = f_{i,j}.$$

We have shown that the truncation error is

$$\tau_{i,j} = \frac{h^2}{12}\left(u_{xxxx}\left(\xi, y_j\right) + u_{yyyy}\left(x_i, \eta\right)\right) \equiv -\nabla_h^2 u\left(x_i, y_j\right) - f_{i,j},$$

where $\xi \in (x_i - h, x_i + h)$ and $\eta \in \left(y_j - h, y_j + h\right)$.

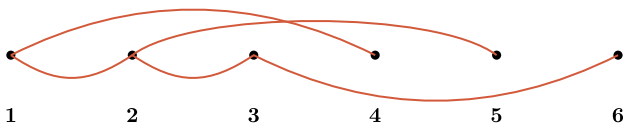


FIGURE 9.9

Graph to illustrate that the matrix $\mathbf{A}$ is strongly connected.

We now define the error in two dimensions, $e_{i,j}$, as

$$e_{i,j} = u\left(x_i, y_j\right) - u_{i,j}. \tag{9.97}$$

Applying the discrete operator to (9.97), we obtain

$$\begin{aligned} -\nabla_h^2 e_{i,j} &= -\nabla_h^2 u\left(x_i, y_j\right) + \nabla_h^2 u_{i,j}, \\ &= \tau_{i,j} + f_{i,j} - f_{i,j} = \tau_{i,j}. \end{aligned}$$

As for the ordinary differential case $-y'' = f$, we need a maximum principle to bound the error, so that the numerical approximation can converge to the true solution.

Given the partial differential equation $L[u] \equiv -\nabla^2 u + \sigma u = f$, where $\sigma \geq 0$, and therefore includes the case where $-\nabla^2 u = f$ $(\sigma = 0)$ and the standard difference approximation $L_h\left[u_{i,j}\right] \equiv -\nabla_h^2 u_{i,j} + \sigma u_{i,j}$, then if $L_h\left[v_{i,j}\right] \leq 0$, then the maximum in $\mathcal{D}$ is bounded above by

$$\max_{\mathcal{D}_h} v_{i,j} \leq \max\left\{0, \max_{\mathcal{C}_h}\left\{v_{i,j}\right\}\right\},$$

where $\mathcal{D}_h$ is the numerical mesh in the interior of the domain and $\mathcal{C}_h$ are the points on the boundary.

For the partial differential equation $-\nabla^2 u = f$, with the corresponding difference scheme, then if $-\nabla_h^2 v_{i,j} \leq 0$ then the maximum of $v_{i,j}$ is less than or equal to zero which is less than the right-hand side bound in the inequality above.

Let $v_{i,j} = \pm e_{i,j} + N g_{i,j}$ where N is the maximum of the truncation error on the interior mesh $\mathcal{D}_h$ and where $g_{i,j} \geq 0$ is again some auxiliary function that will be problem dependent. Applying the numerical approximation to $v_{i,j}$ results in

$$\begin{aligned} -\nabla_h^2 v_{i,j} &= -\nabla_h^2\left(\pm e_{i,j} + N g_{i,j}\right), \\ &= \mp \nabla_h^2 e_{i,j} - N \nabla_h^2 g_{i,j}, \\ &= \pm \tau_{i,j} - N \nabla_h^2 g_{i,j}. \end{aligned}$$

We know that the error cannot be larger than the maximum value of the truncation error, which implies that $-N \leq e_{i,j} \leq N$, which leads to

$$e_{i,j} \leq N \leq N - N \nabla_h^2 g_{i,j} = N\left(1 - \nabla_h^2 g_{i,j}\right) \leq 0,$$

provided $1 - \nabla_h^2 g_{i,j} < 0 \Rightarrow \nabla_h^2 g_{i,j} \geq 1$.

If we now recall the maximum principle mentioned earlier in this section, then we have

$$v_{i,j} \leq \max_{\mathcal{C}_h}\left\{v_{i,j}\right\}.$$

We have that

$$\pm e_{i,j} \leq \pm e_{i,j} + N g_{i,j} = v_{i,j} \leq \max_{\mathcal{C}_h}\left\{0, \max_{\mathcal{C}_h}\left\{\pm e_{i,j} + N g_{i,j}\right\}\right| \leq N \max_{\mathcal{D}_h \cup \mathcal{C}_h}\left\{g_{i,j}\right\}, \tag{9.98}$$

when we have Dirichlet boundary conditions as the error $e_{i,j}$ is equal to zero on the boundary as we have matched the boundary conditions in the numerical scheme. Thus we have

$$\begin{aligned}\left|u\left(x_i, y_j\right) - u_{i,j}\right| &\leq M \max\left\{g_{i,j}\right|, \\ &= \frac{h^2}{12}\left(M_x^4 + M_y^4\right) \max\left\{g_{i,j}\right|.\end{aligned}$$

As before, let us consider an example, as the auxiliary function is problem dependent. Suppose that $\begin{matrix} -\nabla^2 u = f, & (x, y) \in \mathcal{D} \\ u = g, & (x, y) \in \mathcal{C} \end{matrix}$, where $\mathcal{D} = [0, a] \times [0, b]$ for $a \geq b$. As an estimate for the auxiliary function, let us try $g_{i,j} = g\left(x_i, y_j\right)$ where $g\left(x, y\right) = \frac{1}{2}x^2$. This function satisfies $g \geq 0$ and the largest value it can obtain is at $x = a$ and as such the bound of the error is

$$\left|u\left(x_i, y_j\right) - u_{i,j}\right| \leq \frac{h^2}{12}\left(M_x^4 + M_y^4\right) \times \frac{1}{2}a^2, \tag{9.99}$$

provided that $\nabla_h^2 g_{i,j} \geq 1$. If we consider the continuous case then we require $g_{xx} + g_{yy} \geq 1$, which for this choice of auxiliary function we have $1 + 0 \geq 1$. Now we have to check that the discrete operator acting on the auxiliary function satisfies the equivalent condition:

$$\begin{aligned} -\frac{1}{h^2}\left(4g_{i,j} - g_{i-1,j} - g_{i+1,j} - g_{i,j-1} - g_{i,j+1}\right) &\geq 1, \\ -\frac{1}{h^2}\left(2x_i^2 - \frac{1}{2}\left(x_i^2 - 2x_i h + h^2\right) - \frac{1}{2}\left(x_i^2 + 2x_i h + h^2\right) - \frac{1}{2}x_i^2 - \frac{1}{2}x_i^2\right) &\geq 1, \end{aligned}$$

where because of our choice for the auxiliary function not being dependent on y the numerical approximation to the derivative in the y-direction have no change in them at these point on the stencil.

All of the terms in the inequality cancel except for the h^2 terms which, when divided by h^2 and then multiplied by -1, make the left-hand side of the inequality equal to one and hence this auxiliary function is acceptable and we know that the error for this scheme for this specific boundary value problem is bounded by (9.99).

9.6.4 Mixed Boundary Conditions

When we consider differential equations that are functions of multiple dimensions, it is not unusual for the boundary conditions to be functions of both the solution and its outward normal derivative, $\frac{\partial u}{\partial \boldsymbol{n}}$. If we consider the general 2D problem, then we have

$$-\nabla^2 u = f, \quad (x, y) \in \mathcal{D}, \quad \alpha u + \beta \frac{\partial u}{\partial \boldsymbol{n}} = \gamma, \quad (x, y) \text{ on } \mathcal{C}. \tag{9.100}$$

For us to be able to numerically approximate (9.100) we shall extend the notion of ghost, or fictitious, points that are distance h from the boundary, as we did for the numerical approximations to ordinary differential equations. In Fig. 9.10 we have drawn a boundary with its outward normal for the x-direction.

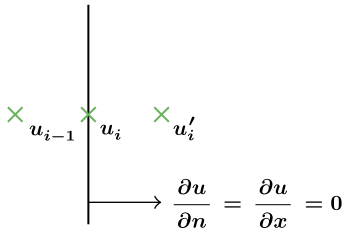


FIGURE 9.10

Illustration of the numerical approximation to the outward normal derivative in the x-direction.

As a result of the illustration in Fig. 9.10, we are able to write down the difference scheme and u_i and u_{i+1}, as well as approximating the boundary condition by a central difference, as

$$\alpha u_{i+1} + \beta \frac{(u_i' - u_i)}{2h} = \gamma, \tag{9.101}$$

where the coefficients α, β, and γ at point $i+1$. We then eliminate u_i' since we can rearrange (9.101) to obtain an expression for u_i' as $u_i' = (\gamma - \alpha u_{i+1})\frac{2h}{\beta} + u_i$. When we have the situation where $\frac{\partial u}{\partial \boldsymbol{n}} = 0$ on the x boundary, this is then implying that x is constant on that boundary. The numerical approximation for this boundary condition would be $\frac{u_i' - u_i}{2h} = 0 \Rightarrow u_i' = u_i$.

We now present the following example to illustrate how to implement the mixed boundary conditions into a numerical approximation to the general Poisson equation. Consider the boundary value problem $-\nabla^2 u = f$ on $[0,1] \times [0,1]$ with $u = 0$ on the x- and y-axes and $\frac{\partial u}{\partial \boldsymbol{n}} = 0$ on the remaining sides of the boundary. For the x-direction the outward normal derivative here is $\frac{\partial u}{\partial x} = 0$ and for the y-direction this boundary condition is $\frac{\partial u}{\partial y} = 0$. We shall set the step size, $h = \frac{1}{2}$, use the natural ordering approach and now derive the associated coefficient matrix.

A schematic of the numerical problem is presented in Fig. 9.11, where the green crosses represent the known values for $u_{i,j}$ and the yellow circles represent the imaginary exterior points. We have also labeled the imaginary points with the corresponding interior point as per the approximation given above for this type of boundary condition.

Applying the five-point stencil, $4u_p - u_N - u_S - u_E - u_W = h^2 f_p = \frac{1}{4} f_p$ to the four interior points leads to the following simultaneous equations:

$$\begin{aligned}
4u_1 \quad -u_2 \quad -u_3 \quad\quad &= \frac{f_1}{4}, \\
-2u_1 \quad +4u_2 \quad\quad -u_4 \quad &= \frac{f_2}{4}, \\
-2u_1 \quad\quad +4u_3 \quad -u_4 \quad &= \frac{f_3}{4}, \\
\quad -2u_2 \quad -2u_3 \quad +4u_4 \quad &= \frac{f_4}{4}.
\end{aligned}$$

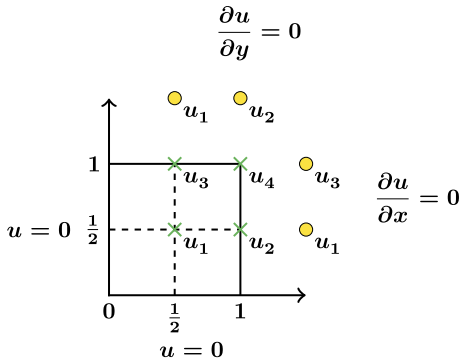


FIGURE 9.11

Schematic of a grid with the extra points for the numerical approximation to the outward normal derivative in the x- and y-directions.

The associated coefficient matrix for the set of equations above is

$$\mathbf{A} = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -2 & 4 & 0 & -1 \\ -2 & 0 & 4 & -1 \\ 0 & -2 & -2 & 4 \end{pmatrix}. \tag{9.102}$$

The first feature that we should note about the matrix $\mathbf{A}$ in (9.102) is that it is no longer symmetric. However, the second feature to notice is that it is strictly diagonally dominant for three of the four rows, and is only diagonally dominant on the last row. It is quite clear that all the points are connected in (9.102) and as such we can say that the matrix $\mathbf{A}$ is a M-matrix, and hence it is invertible. Matrix $\mathbf{A}$ being invertible enables us the five-point stencil for partial differential equation.

If we now decrease the step size to $h = \frac{1}{3}$, then as a result of this we have an increase in the number of points in the interior of the domain from four to nine.

The associated coefficient matrix for $h = \frac{1}{3}$ is

$$\mathbf{A} = \left(\begin{array}{ccc|ccc|ccc} 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & -2 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\ \hline -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -2 & 4 & 0 & 0 & -1 \\ \hline 0 & 0 & 0 & -2 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & -2 & 4 \end{array}\right),$$

where the coefficient matrix has been partitioned to illustrate each row of the interior of the domain. We can clearly see that the matrix is not symmetric and that there are large areas of zeros; specifically there are two blocks that only have zeros in them. Therefore, this is a sparse matrix, but also if we where

to decrease the step size further, which we do not do here, then the coefficient matrix becomes more sparse but is still invertible, as it can easily be shown to be a M-Matrix.

9.7 Self-Adjoint Problem in Two Dimensions

We have already presented methods for the self-adjoint problem in one dimension earlier in this chapter and have seen that it is important to utilize the self-adjoint property to increase the order of the accuracy of the numerical scheme. Therefore, we now extend this theory to two dimensions. We start with the general expression for the self-adjoint problem in two dimensions, which is given by

$$-(pu_x)_x - \left(qu_y\right)_y + ru = f, \tag{9.103}$$

where p, q, r, and f are functions of x, y. As a side note, the problem that we have been considering in the partial differential equations section, $-\nabla^2 u = f$, is a self-adjoint problem.

In the 1D case it was shown that we integrate away the differential operator on the first term in the self-adjoint ordinary differential equation case. Therefore, we wish to be able to utilize the integrability of the partial differential equation in (9.103). We have provided a schematic to illustrate how this is achieved in the 2D case in Fig. 9.12.

The integration-based methods for self-adjoint partial differential equations utilize integrating around the box that has been place around the point of interest in the interior of the domain, while integrating the partial differential equation over the box's interior. Therefore, applying double integration to (9.103), we have

$$\iint_{\blacksquare} -(pu_x)_x - \left(qu_y\right)_y dxdy + \iint_{\blacksquare} rudxdy = \iint_{\blacksquare} fdxdy, \tag{9.104}$$

where the $\blacksquare$ refers to interior of the domain. First, we replace $\iint_{\blacksquare} rudxdy$ by $r_{i,j}u_{i,j}h^2$ and $\iint_{\blacksquare} fdxdy$ by $f_{i,j}h^2$.

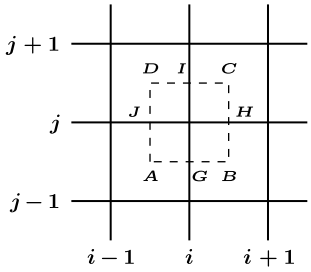


FIGURE 9.12

Schematic of the numerical approximation to the self-adjoint problems.

Before we address the remaining continuous terms in (9.104), we first recall **Green's theorem**, which is

$$\iint_{\blacksquare} P_x - Q_y dxdy = \int_{\square} Pdy + Qdx,$$

where the integral on the left is over the interior of the domain and $\square$ denotes the boundary of the box around the interior point.

For our problem, let $P = -pu_x$ and $Q = qu_y$, and apply Green's theorem above, then we have

$$\int_{\square} -pu_x dy + qu_y dx. \tag{9.105}$$

Here we have to integrate around the box that is illustrated in Fig. 9.12, which implies that we are integrating in the sequence, $A \to B \to C \to D$, which leads to

$$\begin{aligned}\int_{\square} -pu_x dy + qu_y dx &= \int_A^B qu_y dx + \int_B^C -pu_x dy + \int_C^D qu_y dx + \int_D^A -pu_x dy, \\ &= \int_A^B qu_y dx + \int_B^C -pu_x dy - \int_C^D qu_y dx + \int_D^A pu_x dy,\end{aligned} \tag{9.106}$$

where the last two integrals have changed sign due to the direction that we are integrating around the boundary of the box.

The next step is to apply numerical approximations to the integrals in (9.106). The distance between each vertex in our box about the central point is h. Combining this information with an upwind difference scheme for the derivatives in each of the integrals in (9.106) results in the following numerical approximation.

$$q_{i,j-\frac{1}{2}} h \frac{(u_{i,j} - u_{i,j-1})}{h} - p_{i+\frac{1}{2}} h \frac{(u_{i+1,j} - u_j)}{h} - q_{i,j+\frac{1}{2}} h \frac{(u_{i,j+1} - u_{i,j})}{h} + p_{i-\frac{1}{2},j} h \frac{(u_{i,j} - u_{i-1,j})}{h}. \tag{9.107}$$

An important feature to notice about (9.107) is that all of the h terms cancel. Combining (9.107) with the expressions for the other two terms in (9.106) results in the following coefficients for the five points of the stencil:

$u_{i,j-1}$	$q_{i,j-\frac{1}{2}}$
$u_{i-1,j}$	$-p_{i-\frac{1}{2},j}$
$u_{i,j}$	$p_{i+\frac{1}{2},j} + p_{i-\frac{1}{2},j} + q_{i,j-\frac{1}{2}} + q_{i,j+\frac{1}{2}} + h^2 r_{i,j}$
$u_{i+1,j}$	$-p_{i+\frac{1}{2},j}$
$u_{i,j+1}$	$-q_{i,j+\frac{1}{2}}$.

Therefore, the numerical approximation to the 2D self-adjoint problem becomes

$$\begin{aligned}h^2 L_h[u_{i,j}] &= -q_{i,j-\frac{1}{2}} u_S - q_{i,j+\frac{1}{2}} u_N - p_{i+\frac{1}{2},j} u_E - p_{i-\frac{1}{2},j} u_W, \\ &+ \left(p_{i+\frac{1}{2},j} + p_{i-\frac{1}{2},j} + q_{i,j+\frac{1}{2}} + q_{i,j-\frac{1}{2}} + h^2 r_i\right) u_p = h^2 f_p.\end{aligned} \tag{9.108}$$

It is possible to simplify the notation in (9.108) through using the lettering that is present in Fig. 9.12 at the halfway points between the two grid points. As with the other discretizations that we have presented for the boundary value problems it is possible to write the numerical scheme as a matrix-vector equation, where $\mathbf{A}$ is again a symmetric block diagonal matrix with the same structure as for the 1D self-adjoint differential equation. The block matrices for this problem are

$$\mathbf{B} = \begin{pmatrix} & & \\ -p_J & p_h + p_j + q_f + q_g + h^2 f_p & -p_h \\ & & \end{pmatrix},$$

$$\mathbf{C} = \begin{pmatrix} \ddots & & & & \\ & \ddots & & & \\ & & -q_I & & \\ & & & \ddots & \\ & & & & \ddots \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} \ddots & & & & \\ & \ddots & & & \\ & & -q_J & & \\ & & & \ddots & \\ & & & & \ddots \end{pmatrix}.$$

For a standard problem with Dirichlet boundary conditions, this is an M-matrix if $p, q > 0$ and $r \geq 0$.

9.7.1 Solution Methods for Linear Matrix Equations

We have seen through many examples, and from the theory that has been presented in this chapter, that the discretization for boundary value problems lead to a matrix-vector equation, $\mathbf{A}\boldsymbol{u} = \boldsymbol{f}$, where $\mathbf{A}$ is large and sparse, that has to be solved. It is not uncommon for an iterative scheme to be used to solve this type of problem rather than directly inverting the coefficient matrix. We now consider a selection iterative scheme that could be used to solve the matrix equation arising from the different discretizations.

The starting point is to split the matrix $\mathbf{A}$ into a diagonal matrix, $\mathbf{D}$, a strictly lower triangular matrix, $\mathbf{L}$, and a strictly upper triangular matrix, $\mathbf{U}$ as

$$\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{U}.$$

This then enables the matrix equation to be rewritten as

$$(\mathbf{D} - \mathbf{L} - \mathbf{U})\,\boldsymbol{u} = \boldsymbol{f}.$$

9.7.2 Jacobi Method

The Jacobi method is based upon the diagonal matrix $\mathbf{D}$ to precondition the value for the next iteration, while the lower and upper triangular matrices are added together to multiply the previous iteration. This is written mathematically as

$$\begin{aligned} &\mathbf{D}\boldsymbol{u}^{(k+1)} = -(\mathbf{L} + \mathbf{U})\,\boldsymbol{u}^{(k)} = \boldsymbol{f}, \\ \Rightarrow &\boldsymbol{u}^{(k+1)} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})\,\boldsymbol{u}^{(k)} + \mathbf{D}^{-1}\boldsymbol{f}, \quad k = 0, 1, \ldots, \end{aligned} \tag{9.109}$$

where $\boldsymbol{u}^{(0)}$ is the initial guess for the solution. The matrix $\mathbf{B} = \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$ is the **Jacobi iteration matrix**. Applying the factorization for the Jacobi method from (9.109) results in the iteration equation for the five-point stencil as

$$\begin{aligned} h^2 g_p &= -u_N^{(k)} - u_S^{(k)} + 4u_p^{(k+1)} - u_W^{(k)} - u_E^{(k)}, \\ \Rightarrow u_p^{(k+1)} &= \frac{1}{4}\left(u_N^{(k)} + u_S^{(k)} + u_W^{(k)} + u_E^{(k)} + h^2 q_p\right), \end{aligned}$$

for $k = 0, 1, \ldots$.

9.7.3 Gauss-Seidel

The Gauss-Seidel iteration scheme is based on a different combination of the factorization of $\mathbf{A}$. For this iterative scheme we have the diagonal plus the lower triangular matrix preconditioning the new iterate, while the upper triangular still multiplies the previous iterate. Mathematically the Gauss-Seidel iterative scheme is defined as

$$\begin{aligned} (\mathbf{D} - \mathbf{L})\,\boldsymbol{u}^{(k+1)} &= \mathbf{U}\boldsymbol{u}^{(n)} + \boldsymbol{f}, \\ \Rightarrow \boldsymbol{u}^{(k+1)} &= (\mathbf{D} - \mathbf{L})^{-1}\,\mathbf{U}\boldsymbol{u}^{(n)} + (\mathbf{D} - \mathbf{L})^{-1}\boldsymbol{f}. \end{aligned} \tag{9.110}$$

The matrix $\mathcal{L}_1 = (\mathbf{D} - \mathbf{L})^{-1}\,\mathbf{U}$ is the **Gauss-Seidel iteration matrix**. Applying this iteration scheme to the five-point stencil results in

$$\boldsymbol{u}^{(k+1)} = \frac{1}{4}\left(u_N^{(k)} + u_E^{(k)} + u_W^{(k+1)} + u_S^{(k+1)} + h^2 q_p\right). \tag{9.111}$$

9.7.4 Successive Over-Relaxation Method

The successive over-relaxation (SOR) method iteration comprises of two parts that have different weights as

$$u_p^{(k+1)} = \omega\left(u_p^{(k+1)}\right)_{GS} + (1 + \omega)\,u_p^k, \tag{9.112}$$

where the subscript GS refers to Gauss-Seidel iteration matrix. When the parameter $\omega = 1$, the iteration scheme in (9.112) becomes the Gauss-Seidel iteration scheme. When the Gauss-Seidel method does converge, then for better convergence we would use $\omega > 1$. There is always the question of whether there is an optimum value for ω to ensure convergence.

Applying the SOR method to the five-point stencil, we have

$$u_p^{(k+1)} = \frac{\omega}{4}\left(u_N^{(k)} + u_W^{(k+1)} + u_E^{(k)} + u_S^{(k+1)} + h^2 g_p\right) + (1 - \omega)\,u_p^{(k)}. \tag{9.113}$$

The general matrix expression for iterative scheme from (9.112) is given by

$$\boldsymbol{u}^{(k+1)} = (\mathbf{D} - \omega\mathbf{L})^{-1}\left[(1 - \omega)\,\mathbf{D} + \omega\mathbf{U}\right]\boldsymbol{u}^{(k)} + (\mathbf{D} - \omega\mathbf{L})^{-1}\boldsymbol{f}. \tag{9.114}$$

Finally the **SOR iteration matrix** is defined as $\mathcal{L}_\omega = (\mathbf{D} - \omega\mathbf{L})^{-1}\left[(1 - \omega)\,\mathbf{D} + \omega\mathbf{U}\right]$.

The reason we have defined the iteration matrices for these three iterative schemes is to be able to present the following five theorems that provide properties for the convergence of the schemes.

Theorem 9.8. *For an iterative scheme with iteration matrix* $\boldsymbol{C}$*, then the iteration scheme converges to the solution of* $\boldsymbol{Au} = \boldsymbol{f}$ *from any starting value* $\boldsymbol{u}^{(0)} \Leftrightarrow \rho(\boldsymbol{C}) < 1$.

Theorem 9.9. *If the matrix* $\boldsymbol{A}$ *is strictly diagonally dominant, then the Jacobi and Gauss-Seidel iteration schemes converge.*

Theorem 9.10. *If the SOR converges, then the parameter* ω *satisfies* $0 < \omega < 2$*. Note: This is only a necessary condition.*

Theorem 9.11. *For a symmetric matrix* $\boldsymbol{A}$ *then the SOR iteration scheme will converge for all* $\omega \in (0, 2) \Leftrightarrow \boldsymbol{A}$ *is positive definite.*

Theorem 9.12. *If the matrix* $\boldsymbol{A}$ *is symmetric, strictly diagonally dominant, and all of the diagonal entries satisfy* $a_{i,i} > 0$*, then* $\boldsymbol{A}$ *is positive definite* $\Leftrightarrow$ *the SOR iterative scheme converges* $\forall \omega \in (0, 2)$.

We shall not provide proofs of these theorems here, but use them to help identify what is required from your application of the iterative schemes to guarantee convergence.

Now we shall introduce the definition of a matrix being 2-cyclic, which becomes important in determining if certain iterative schemes will converge to solve the matrix-vector equation, $\mathbf{Au} = \boldsymbol{f}$.

Definition 9.13. An $N \times N$ matrix $\mathbf{A}$ is said to be 2-cyclic if the following property holds: $\exists$ disjoint sets S and T of the first positive integers such that the intersection of the two sets, $S \cap T = \varnothing$, and the union of the two sets, $S \cup T = \{1, 2, \ldots, N\}$, where for

$$a_{ij} \neq 0 \Rightarrow \begin{cases} l = m, \\ \text{or } i \in S \text{ and } j \in T, \\ \text{or } i \in T \text{ and } j \in S. \end{cases} \tag{9.115}$$

To help illustrate how to determine if a matrix is 2-cyclic, we return to the boundary value problem, $-\nabla^2 u = g$ with Dirichlet boundary conditions. Applying the five-point approximation for the 4×4 interior point mesh for the unit square domain results in the familiar matrix

$$\mathbf{A} = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}.$$

We now determine if there are two disjointed sets for the integers of the indices for the non-zero entries of $\mathbf{A}$:

$$\begin{array}{llll} a_{12} \neq 0 & 1 \in S & 2 \in T & \\ a_{13} \neq 0 & 1 \in S & 3 \in T & \\ a_{21} \neq 0 & 2 \in T & 1 \in S & \\ a_{24} \neq 0 & 2 \in T & 4 \in S & \Rightarrow \quad \begin{array}{l} S = \{1, 4\} \\ T = \{2, 3\} \end{array}, \quad \therefore \mathbf{A} \text{ is 2-cyclic.} \\ a_{31} \neq 0 & 3 \in T & 1 \in S & \\ a_{34} \neq 0 & 3 \in T & 4 \in S & \\ a_{42} \neq 0 & 4 \in S & 2 \in T & \\ a_{43} \neq 0 & 4 \in S & 3 \in T & \end{array}$$

The next definition that we shall introduce is that for the **consistently ordered** property of 2-cyclic matrices, which is given by:

Definition 9.14. A $N \times N$ 2-cyclic matrix $\mathbf{A}$ is said to be **consistently ordered** if the following holds: $\exists$ disjoint sets S_k such that $\bigcup_k S_k \{1, 2, \ldots, N\}$, where for

$$a_{ij} \neq 0 \Rightarrow \begin{cases} i = j, \\ i < j \quad i \in S_k \quad j \in S_{k+1}, \\ i > j \quad i \in S_k \quad j \in S_{k-1}. \end{cases} \tag{9.116}$$

Returning to the example above, then the associated sets are

$$\begin{array}{lll} a_{12} & 1 \in S_1 & 2 \in S_2 \\ a_{13} & 1 \in S_1 & 3 \in S_2 \\ a_{21} & 2 \in S_2 & 1 \in S_1 \\ a_{24} & 2 \in S_2 & 4 \in S_3 \\ a_{31} & 3 \in S_2 & 1 \in S_1 \\ a_{34} & 3 \in S_2 & 4 \in S_3 \\ a_{42} & 4 \in S_3 & 2 \in S_2 \\ a_{43} & 4 \in S_3 & 3 \in S_2 \end{array} \Rightarrow S_1 = \{1\}, \; S_2 = \{2, 3\}, \; S_3 = \{4\},$$

and therefore the matrix $\mathbf{A}$ for our example is also consistently ordered.

Given the two definitions above, it is now possible to state Theorem 9.15.

Theorem 9.15. *A 2-cyclic matrix $\boldsymbol{A}$ that can be expressed in the form $\boldsymbol{A} = (\boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U})$ is consistently ordered if all of the eigenvalues of the matrix $\boldsymbol{K}_\alpha$, where*

$$\boldsymbol{K}_\alpha \equiv \boldsymbol{D}^{-1} \left[\alpha \boldsymbol{L} + \frac{1}{\alpha} \boldsymbol{U} \right] (\alpha \neq 0), \tag{9.117}$$

are independent of α, $\forall \alpha \in \mathbb{C}$.

Again returning to the five-point stencil example, then the associated $\mathbf{K}_\alpha$ matrix as defined in (9.117) is

$$\mathbf{K}_\alpha = \begin{pmatrix} 0 & \frac{1}{4\alpha} & \frac{1}{4\alpha} & 0 \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ 0 & \frac{\alpha}{4} & \frac{\alpha}{4} & 0 \end{pmatrix}.$$

We now introduce a theorem associated with properties relating eigenvalues of different iteration matrices to the eigenvalues of the Jacobi iteration matrix.

Theorem 9.16. *If the matrix $\boldsymbol{A}$ is consistently ordered, then the eigenvalues of the associated $\boldsymbol{K}_\alpha$ matrix are the eigenvalues of $\boldsymbol{K}_1$, which is to say the eigenvalues of*

$$\left(\boldsymbol{D}^{-1} \boldsymbol{L} + \boldsymbol{D}^{-1} \boldsymbol{U} \right) = \boldsymbol{D}^{-1} (\boldsymbol{L} + \boldsymbol{U}),$$

where the iteration matrix on the right-hand side above is that of the Jacobi iterative scheme.

Proof. To prove this theorem, we require the similarity transformation which, to recap, is that if there exists a matrix $\mathbf{G}_\alpha$ such that $\mathbf{K}_\alpha = \mathbf{G}_\alpha^{-1} \mathbf{B} \mathbf{G}_\alpha^{-1}$, then the eigenvalues of $\mathbf{K}_\alpha$ are equal to the eigenvalues of $\mathbf{B}$. This is to say,

$$
\begin{aligned}
0 = |\mathbf{K}_\alpha - \lambda \mathbf{I}| &= \left|\mathbf{G}_\alpha \mathbf{B} \mathbf{G}_\alpha^{-1} - \lambda \left(\mathbf{G}_\alpha \mathbf{G}_\alpha^{-1}\right)\right|, \\
&= \left|\mathbf{G}_\alpha \left(\mathbf{B} - \lambda \mathbf{I}\right) \mathbf{G}_\alpha^{-1}\right|, \\
&= \left|\mathbf{G}_\alpha\right| \left|\mathbf{G}_\alpha^{-1}\right| \left|\mathbf{B} - \lambda \mathbf{I}\right|, \\
&= \left|\mathbf{B} - \lambda \mathbf{I}\right| = 0.
\end{aligned}
$$

Therefore, this proves that if we can find a similarity transform for $\mathbf{K}_\alpha$ to the iterative matrix of the Jacobi iterative scheme then the eigenvalues of $\mathbf{K}_\alpha$ will be those of the Jacobi iteration matrix, which is independent of α and will have proven both of the last two theorems. We therefore need to show that if

$$
\mathbf{K}_\alpha = \mathbf{G}_\alpha \mathbf{B} \mathbf{G}_\alpha^{-1} \equiv \mathbf{G}_\alpha \mathbf{D}^{-1} \left(\mathbf{L} + \mathbf{U}\right) \mathbf{G}_\alpha^{-1}, \tag{9.118}
$$

then the eigenvalues of $\mathbf{K}_\alpha$ are those of $\mathbf{B}$.

Suppose that the eigenvalues of $\mathbf{K}_\alpha$ are λ, then we have

$$
\begin{aligned}
&\mathbf{K}_\alpha \boldsymbol{v} = \lambda \boldsymbol{v}, \\
\Rightarrow &\mathbf{G}_\alpha \mathbf{B} \mathbf{G}_\alpha^{-1} \boldsymbol{v} = \lambda \boldsymbol{v}, \\
\Rightarrow &\mathbf{B} \mathbf{G}_\alpha^{-1} \boldsymbol{v} = \lambda \mathbf{G}_\alpha^{-1} \boldsymbol{v}, \\
\Rightarrow &\mathbf{B} \boldsymbol{w} = \lambda \boldsymbol{w}, \quad \text{where } \boldsymbol{w} = \mathbf{G}_\alpha^{-1} \boldsymbol{v}.
\end{aligned}
$$

Example. *We now return once more to the five-point stencil example, where we have* $\boldsymbol{A}$ *defined by*

$$
\boldsymbol{A} = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}.
$$

It has already been shown that $\boldsymbol{A}$ *is* 2*-cyclic and consistently ordered. Forming the three matrices,* $\mathbf{D}$*,* $\mathbf{L}$*, and* $\mathbf{U}$*, we have*

$$
\boldsymbol{D} = \begin{pmatrix} 4 & & & \\ & 4 & & \\ & & 4 & \\ & & & 4 \end{pmatrix}, \quad \boldsymbol{L} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}, \quad \boldsymbol{U} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}.
$$

We have also shown that the $\boldsymbol{K}_\alpha$ *matrix is*

$$
\boldsymbol{K}_\alpha = \begin{pmatrix} 0 & \frac{1}{4\alpha} & \frac{1}{4\alpha} & 0 \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ 0 & \frac{\alpha}{4} & \frac{\alpha}{4} & 0 \end{pmatrix}.
$$

The aim now is to find the matrix $\boldsymbol{G}_\alpha$ *such that* $\boldsymbol{K}_\alpha = \boldsymbol{G}_\alpha \boldsymbol{B} \boldsymbol{G}_\alpha^{-1}$. *Therefore, the question is: What is* $\boldsymbol{G}_\alpha$? *For a general tri-diagonal block matrix from natural ordering, the required* $\boldsymbol{G}_\alpha$ *matrix is*

$$\boldsymbol{G}_\alpha = \left(\begin{array}{ccc|ccc|ccc} \alpha & & & & & & & & \\ & \alpha^2 & & & & & & & \\ & & \alpha^3 & & & & & & \\ \hline & & & \alpha^2 & & & & & \\ & & & & \alpha^3 & & & & \\ & & & & & \alpha^4 & & & \\ \hline & & & & & & \alpha^3 & & \\ & & & & & & & \alpha^4 & \\ & & & & & & & & \alpha^5 \end{array}\right),$$

where all the empty areas in the matrix above are filled with zeros.

For a $M = N \times P$ *problem, where* N *is the number of horizontal grid points per row, and* P *is the total number of points in the* y*-direction, then there are* P *block diagonal matrices along the diagonal of* $\boldsymbol{G}_\alpha$ *that are* $N \times N$ *in size and have the power of* α^i *for the first entry of the diagonal of that block matrix equivalent to the number of that block on the larger diagonal, where* $i = 1, 2, \ldots, P$. *For example, if we had an* $M = 3 \times 4 = 12$ *where we have four vertical rows of length* 3 *in the horizontal, then there would be four blocks of diagonal matrices that are* 3×3 *and the increasing powers of* α *would start from* α, α^2, α^3, *and* α^4, *respectively.*

Returning to our example, we have $M = 2 \times 2 = 4$ *total points, and thus our block diagonal matrices will be* 2 *by* 2 *as*

$$\boldsymbol{G}_\alpha = \left(\begin{array}{cc|cc} \alpha & & & \\ & \alpha^2 & & \\ \hline & & \alpha^2 & \\ & & & \alpha^3 \end{array}\right) \quad \textit{therefore} \quad \boldsymbol{G}_\alpha^{-1} = \begin{pmatrix} \alpha^{-1} & & & \\ & \alpha^{-2} & & \\ & & \alpha^{-2} & \\ & & & \alpha^{-3} \end{pmatrix}.$$

Applying our two matrices above to $\boldsymbol{B}$, *we have*

$$\begin{aligned} \boldsymbol{G}_\alpha \boldsymbol{B} \boldsymbol{G}_\alpha^{-1} &= \boldsymbol{G}_\alpha \begin{pmatrix} 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{4} & 0 \end{pmatrix} \boldsymbol{G}_\alpha^{-1}, \\ &= \begin{pmatrix} 0 & \frac{\alpha}{4} & \frac{\alpha}{4} & 0 \\ \frac{\alpha 2}{4} & 0 & 0 & \frac{\alpha^2}{4} \\ \frac{\alpha^2}{4} & 0 & 0 & \frac{\alpha^2}{4} \\ 0 & \frac{\alpha^3}{4} & \frac{\alpha^3}{4} & 0 \end{pmatrix} \boldsymbol{G}_\alpha^{-1}, \end{aligned}$$

$$= \begin{pmatrix} 0 & \frac{1}{4\alpha} & \frac{1}{4\alpha} & 0 \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ \frac{\alpha}{4} & 0 & 0 & \frac{1}{4\alpha} \\ 0 & \frac{\alpha}{4} & \frac{\alpha}{4} & 0 \end{pmatrix} = \boldsymbol{K}_\alpha,$$

which then implies that $\boldsymbol{A}$ *is consistently ordered.*

Given all of the theorems and definitions presented so far in this section, it is now possible to state a theorem that quantifies the conditions to ensure optimal rate of convergence for the SOR methods as follows.

Theorem 9.17. *If the matrix* $\boldsymbol{A}$ *is 2-cyclic and consistently ordered, then the eigenvalues of the Jacobi iteration matrix* $\boldsymbol{B}^{-1}$ *are in* $(-1, 1)$*, which then implies that the Jacobi iteration method will converge and the SOR iteration method will converge for all* $\omega \in (0, 2)$*, which includes the Gauss-Seidel iteration method, which is when* $\omega = 1$.

Moreover, the optimum rate of convergence occurs when

$$\omega = \omega_{opt} = \frac{2}{1 + \sqrt{1 - (\rho(\boldsymbol{B}))^2}}, \tag{9.119}$$

where $\rho(\boldsymbol{B})$ *is the spectral radius of the matrix* $\boldsymbol{B}$ *which is the maximum in modulus of the eigenvalues of* $\boldsymbol{B}$.

An interesting property of the spectral radius of $\mathbf{B}$ is that $\rho(\mathbf{B}) = \rho\left(\mathcal{L}^2\right)$, which is the spectral radius of the iteration matrix of the Gauss-Seidel method.

Exercise 9.18. *Consider the differential equation* $-\nabla^2 u = 0$ *on the domain* $\{(x, y), 0 \le x \le 4, 0 \le y \le 3\}$ *with* $u = 0$ *for* $x = 0, 4$ *and* $y = 0, 3$. *Using a mesh spacing of* $h = 1$ *for both directions, write down the difference equations in matrix form using the four orderings: natural, red/black, diagonal, and counter-clockwise.*

For each case, investigate whether the graph of the coefficient matrix is strongly connected, whether the matrix is 2-cyclic, and whether the matrix is consistently ordered. For the first two orderings mentioned above, show that there exists a matrix $\mathbf{G}_\alpha$ *such that* $\mathbf{K}_\alpha - \alpha\mathbf{D}^{-1}\mathbf{L} + \frac{1}{\alpha}\mathbf{D}^{-1}\mathbf{U}$ *can be written as* $\mathbf{K}_\alpha = \mathbf{G}_\alpha \mathbf{B} \mathbf{G}_\alpha^{-1}$, *where* $\mathbf{B} \equiv \mathbf{D}^{-1}(\mathbf{L} + \mathbf{U})$.

Exercise 9.19. *Consider the solution of* $-\nabla^2 u = 1$ *on the unit square* $0 \le x, y \le 1$ *with the boundary conditions,* $u(t, 0) = u(0, t)$, $u_y(t, 0) = -u_x(0, t)$, $u(t, 1) = u(1, t) = 1$ *for* $0 \le t \le 1$. *Using a uniform mesh of interval* $h = \frac{1}{2}$, *derive the difference equation obtained with the standard five-point difference scheme to yield a symmetric* 3×3 *coefficient matrix. Show that the matrix is non-singular, that the Jacobi iteration and the SOR iteration converges, and determine the optimum acceleration parameter for the SOR iteration.*

9.8 Periodic Boundary Conditions

In the derivations that we have considered so far, we have addressed boundary conditions that are a specific value or function for u on the boundary, or boundaries in the partial differential equations case, and mixed boundary conditions which include derivatives across the boundary, or conditions on the outward normal derivative on a specific boundary. However, in geophysical modeling it is possible that the

boundary may not be a physical barrier, which means that the flow can return to its starting point. Examples of physical flows with this **periodic** boundary are the weather systems in the atmosphere, ocean currents as well. Therefore, we need to address how to numerical approximate this type of boundary condition.

When we were considering Neumann or mixed boundary conditions for both the ordinary and partial differential equations, we introduced ghost, imaginary, points exterior to the domain and the boundary, which enables us to approximate the derivative on the boundary through a central differencing scheme. These central differences enable us to write the ghost point in terms of known interior or boundary point. For periodic flow, where the domain wraps around to meet itself, we employ a similar approach to the ghost point technique, but this time the points are still in the interior.

We consider a simple 1D advection problem which has continuous partial differential equation

$$\frac{\partial b}{\partial t} + c\frac{\partial b}{\partial x} = f(x,t), \tag{9.120}$$

where c is the wind speed and b is buoyancy. This equation is part of the Eady model. We set the boundaries in the x-direction to be at 0 and 1 where there are N points, and the distance between each point is Δx, such that $N\Delta x = 1$. For the time derivative we have a time step of size Δt. If we apply a central difference to the spatial derivative and an upwind scheme for the time component, then the discrete approximation to (9.120) becomes

$$\frac{b_i^{n+1} - b_i^n}{\Delta t} + c\frac{b_{i+1}^n - b_{i-1}^n}{2\Delta x} = f_i^n. \tag{9.121}$$

If we consider the point at $i = 1$, then we require a point at $i = 0$ as part of (9.121) but this point does not exist with this index number. However, the domain loops round back to $x = 0$ and as such we do have a point at $i = -1$ but it is $i = N$. Therefore, applying the continuous periodic boundary condition results in the equivalent discrete condition at $x = 0$ of $b_{0-1}^n = b_N^n$. We also have the same problem when we arrive at $x = 1$, where now it is not $i - 1$ point that concerns us but the $i + 1$ point at $i = N$. As we know that the boundary is looping round therefore at the $i = N$ point, the $N + 1$ point is at $i = 1$ point. Therefore, the numerical periodic boundary condition at $x = 1$ is $b_{N+1}^n = b_1^n$.

Therefore, the discrete equations for all the grid points in the domain are

$$b_1^{n+1} = b_1^n - \frac{c\Delta t}{2\Delta x}\left(b_2^n - b_N^n\right) + \Delta t f_1^n, \tag{9.122a}$$

$$b_i^{n+1} = b_i^n - \frac{c\Delta t}{2\Delta x}\left(b_{i+1}^n - b_{i-1}^n\right) + \Delta t f_i^n, \quad \text{for } i = 2, \ldots, N-1, \tag{9.122b}$$

$$b_N^{n+1} = b_N^n - \frac{c\Delta t}{2\Delta x}\left(b_1^n - b_{N-1}^n\right) + \Delta t f_N^n. \tag{9.122c}$$

It is possible to express (9.122a)–(9.122c) as a matrix vector equation where to help illustrate the affects periodic boundary conditions have on the matrix-vector equation we have taken $N = 5$, which results in

$$\begin{pmatrix} b_1^{n+1} \\ b_2^{n+1} \\ b_3^{n+1} \\ b_4^{n+1} \\ b_5^{n+1} \end{pmatrix} = - \begin{pmatrix} 1 & \mu & 0 & 0 & -\mu \\ -\mu & 1 & \mu & 0 & 0 \\ 0 & -\mu & 1 & \mu & 0 \\ 0 & 0 & -\mu & 1 & \mu \\ \mu & 0 & 0 & -\mu & 1 \end{pmatrix} \begin{pmatrix} b_1^n \\ b_2^n \\ b_3^n \\ b_4^n \\ b_5^n \end{pmatrix} + \Delta t \begin{pmatrix} f_1^n \\ f_2^n \\ f_3^n \\ f_4^n \\ f_5^n \end{pmatrix}, \tag{9.123}$$

where $\mu = \frac{c\Delta t}{2\Delta x}$.

If we did not have the time component in the continuous partial differential equation but instead were solving $\frac{\partial^2 \psi}{\partial x^2} = Q$ in the interior of a domain that had periodic boundary conditions in the horizontal direction with a central difference approximation, then we would have to solve the following matrix-vector equation. Again we take $N = 5$:

$$\begin{pmatrix} -2 & 1 & 0 & 0 & 1 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 1 & 0 & 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_5 \end{pmatrix} = \begin{pmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \\ Q_5 \end{pmatrix}. \tag{9.124}$$

The first feature to notice about the matrix in (9.124) is that it is not strictly dominant for at least one row. Another feature to notice is that the matrix is not 2-cyclic either; that is to say, we cannot form disjointed sets such that the indices cover all the integers to 5. This is indicating that there could be a problem with this formulation. In fact, the matrix above is singular. If you were to add all the rows from $i = 2, \ldots, 5$ to the first row, you would obtain a row of zeros. Thus the continuous problem is ill-posed and the resulting discretization results in a singular matrix.

There are technique to overcome the problem of ill-posedness of the Laplacian, and the Poisson equation with periodic domains. The first technique is to assign a specific value to a point in the interior, or on the boundary, which acts as a constraint on the problem. A second approach is to add a small perturbation to the diagonal entries to make the matrix strictly diagonally dominant, but we have to be cautious here as the original matrix is singular and there can be a lot of sensitivity to perturbing away from a singular matrix. A third technique is to introduce some energy measure that ensures that perhaps mass is conserved, which makes the matrix non-singular.

9.9 Summary

In this chapter we have extended numerical approximations theory for ordinary and partial differential equations from the initial value problems to the boundary value problems. The first noticeable difference between the numerical approaches to the two different types of differential equations is that the boundary value problems result in a matrix-vector equation to invert, while for the time dependent initial value problems we generated a sequence for the explicit methods, but did have matrices to invert when considering implicit methods.

We have derived the truncation error for different numerical approximation depending on the type of boundary conditions that we are considering: Dirichlet, Neumann, or mixed. We did not do this for periodic boundary conditions, but as we saw the numerical approximations can lead to a singular

matrix. We have introduce the definition of 2-cyclic, strongly connected, consistently order properties of M-matrices which are conditions that enable us to know if the matrix from the numerical approximation is invertible.

We have introduced three different iteration schemes to invert the large, sparse matrices that arise from the five-point stencil to approximate the Laplace or the Poisson equation. These schemes were the Jacobi, Gauss-Seidel, and SOR methods.

In the next chapter we move on to the third numerical modeling chapter in this book, where now we focus on a specific type of differential equation, the **advection** equation, and a specific class of approximation to these differential equations: the **semi-Lagrangian methods**.

CHAPTER

Introduction to Semi-Lagrangian Advection Methods

10

Contents

10.1 History of Semi-Lagrangian Approaches

In the previous chapters on numerical modeling we have considered the advection, wave, equation, in its Eulerian form, but we could also consider the flow in a Lagrangian formulation, and then discretize this formulation of the advection equation. Before we introduce the theory, we shall review the different developments of the approximations for the semi-Lagrange approach.

We start with a summary of the introduction of Bates and McDonald's 1982 paper [27]; the latter author has published many papers on the development of semi-Lagrangian schemes for numerical weather prediction in both Cartesian and spherical coordinates, for multiple dimensions, as well as for

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00014-0

different orders of time discretizations, along with different approximations to the forcings. The authors indicate that this approach was first published with an application in meteorology in 1952 [126]. In [126] a graphical method was applied to solve the barotropic vorticity equation with a 24 h time step.

The next development of the theory that leads to the semi-Lagrangian approach came in a 1955 paper by Welander [469], where it is shown that if we consider the case of a set of fluid particles that are initially regularly distributed, they soon become greatly deformed and are then rendered unsuitable for numerical integration. The first indication of a semi-Lagrangian approach for advection comes in [476], where it is referred to as quasi-Lagrangian, where the authors' approach still focuses on movement of particles but now they focus on the set of particles that arrived at a regular set of grid point and then trace **backwards** over a single time interval, Δt, to their ***departure points*** at the previous time. It is not always the case that the departure points will coincide with a grid point, and as such the values of the tracer, or dynamical quantities, at neighboring grid points are interpolated to the departure points. Therefore, the reason why this approach is referred to as *semi-Lagrangian* is that in the true Lagrangian setting, the set of particles change at every time step, whereas in the semi-Lagrangian approach, we follow the set of particles that arrive at a grid point.

Throughout the 1960s and 1970s, the semi-Lagrangian approach was applied to many different numerical prediction models as well as analyses of the properties of the semi-Lagrangian schemes; a summary of these applications can be found in [27]. However, it is the property of the semi-Lagrangian schemes demonstrated in [368] that made the semi-Lagrangian schemes appealing to operational numerical weather, ocean, and hydrological prediction. The property that was demonstrated in [368] was that the semi-Lagrangian approach, when compared to the equivalent Euler formulation of advection, could use significantly larger time steps than the Eulerian formulation and remain numerically stable. This was demonstrated in a divergent barotropic model with a consistent spatial resolution for the time period, but with a time step greater than or equal to an hour.

As indicated in [27], the underlying assumption for the semi-Lagrangian approach in [368] is to interpolate to the departure point using the grid points surrounding the departure point. This implies that when the advecting winds are strong, the departure point could be many spatial grid intervals, Δx, upstream from the arrival grid point. In [27] the authors call an approach where the interpolation points for the departure points are many gird intervals away from the arrival point a **multiply upstream** scheme.

The focuses of the Bates and McDonald's (1982) paper are; first to verify that the scheme proposed in [368] is stable, which we shall derive in later, secondly to determine damping properties of the semi-Lagrangian scheme, and finally to extend the semi-Lagrangian approach to the horizontal advection term of a multilevel primitive equation model, to show that it is possible to obtain faster numerical integrations of these equations than was possible with the Eulerian formulations.

An important advancement from [27] is the introduction of a two-dimensional version of a semi-Lagrangian approach. Thus for a bi-linear interpolation scheme, they use four surrounding grid points to form a box around the departure point to obtain the value of the dynamical quantity at the departure point. They also introduce the bi-quadratic interpolation scheme, which requires nine grid points; eight of the points form a box around the departure point and then the last grid point is the nearest one in the interior of the box.

Another important result from [27] was that the stability criteria for the bi-interpolation schemes was equal to the product of the stability criterion in the two different spatial directions. Therefore, if

using an equally spaced grid/mesh to approximate the flow, then we only need to derive the stability criterion in one direction and then ensure that both directions satisfy that stability criterion.

In the [279] paper, titled "Accuracy of multiply-upstream, semi-Lagrangian advection schemes," we are introduced to the bi-cubic and bi-quartic interpolation schemes, and are shown their stability criteria. As we mentioned with the bi-quadratic interpolation scheme, we require nine grid points to obtain an approximation to the value of the tracer at the departure point; for the bi-cubic interpolation approach we require 16 grid points to form the interpolation; and for the bi-quartic interpolation we need 25 grid points. It is clear that obtaining a more accurate estimate of the tracer at the departure point requires a swift increase in the number of surrounding grid points. Remember, this is still only in two dimensions.

McDonald published another ground-breaking paper on semi-Lagrangian methods in 1986 [280], which became the foundation for most of the world's operational numerical weather prediction centers' implementations of semi-Lagrangian advection. The McDonald 1986 paper "A Semi-Lagrangian and semi-implicit two time-level integration scheme" introduces the use of mid-points for the back trajectory calculations, but also the semi-implicit approximation to the forcing term.

In the [284] paper titled "Semi-Lagrangian integration of a gridpoint shallow water model on the sphere," as the title suggests, the semi-Lagrangian approach was extended to spherical coordinates; we shall go into more detail about numerical modeling on a sphere in Chapter 12. As well as the work by McDonald, there was also a lot of work being undertaken by Purser [354] and by Staniforth and Côté [160,411,428].

Even to this day there is still much ongoing research into making the semi-Lagrangian approaches more efficient and/or more accurate by taking higher-order approximations to the temporal derivative, trying to avoid interpolation by using different approximations to obtain the departure point and the value of the trace at that point. Given this brief overview of the history of the development for the semi-Lagrangian approach for numerical weather prediction, it should be noted that semi-Lagrangian techniques are used in many geoscience fields, not just meteorology. Semi-Lagrangian advection approximations are used extensively in hydrological modeling of advected-dominated river systems [494], as well as in ocean modeling [227]. Given this history of some of the development of semi-Lagrangian theory, we now move on to introduce some of the theory of semi-Lagrangian schemes.

10.2 Derivation of Semi-Lagrangian Approach

The starting point for the derivation of the semi-Lagrangian numerical schemes is to describe the advection equation in terms of the Lagrange derivative as

$$\frac{D\psi}{Dt} = f\left(x,t\right), \tag{10.1}$$

which is the equivalent to the Eulerian form

$$\frac{\partial\psi}{\partial t} + \bar{u}\frac{\partial\psi}{\partial x} = f\left(x,t\right), \tag{10.2}$$

through the definition of total derivatives

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \frac{dx}{dt}\frac{\partial}{\partial x},$$

and the definition of velocity

$$\frac{dx}{dt} = u.$$

We now have to consider how to discretize (10.1). We have seen in Chapter 8 that when we have a time component, we can discretize using an upwind scheme, a central difference approximation, or a multistep upwind scheme. These same approaches can be applied (10.1). We shall consider the first-order upwind scheme approximation to the time derivative in (10.1) given by

$$\frac{D\psi}{Dt} \approx \frac{\tilde{\psi}\left(x_j, t^{n+1}\right) - \tilde{\psi}\left(\tilde{x}_j^n, t^n\right)}{\Delta t} = f_{i,j}^n, \tag{10.3}$$

where $\tilde{\psi}$ is the approximation to the tracer ψ at $\tilde{x}_j^n$, which is the departure point of a trajectory originating at time t^n that arrives at the grid point $\left(x_j, t^{n+1}\right)$. Recalling from earlier, we are assuming that the value of the tracer at the arrival point is equal to the value of the tracer at the departure point.

To find the departure point we must travel backwards following the characteristics; this movement backwards is referred to as the **backwards trajectory**.

The first situation we consider is the case where the forcing term, the right-hand side function in (10.1), is zero. We shall also consider the case where the advection wind is constant with respect to x and t. Given these situations, the backward trajectory calculation to the departure point, $\tilde{x}_j^n$, is given by

$$\tilde{x}_j^n = x_j^{n+1} - \overline{u}\Delta t. \tag{10.4}$$

In Fig. 10.1 we have presented a schematic of the back trajectory approach. An important feature to note about this figure is that it is likely that the departure point at time t^n does not coincide with a grid point that arrives at our grid point at t^{n+1}. When the departure point does not coincide with a grid point, then we need to define a distance from the nearest grid point downwind to the direction of the characteristic.

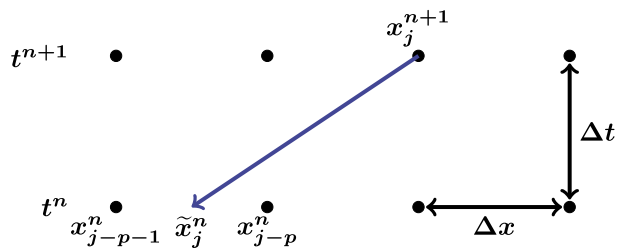


FIGURE 10.1

Schematic of the back trajectory.

Given that our departure point, $\tilde{x}_j^n$, is not at a grid point, then we have to use some form of interpolation to obtain a value for ψ at the departure point; because when there is no forcing term, the value for ψ at the departure point becomes the value for ψ at the grid point at time t^{n+1}.

The first step in a semi-Lagrangian algorithm is to find the integer p of the grid point that is the nearest, downwind or upwind, to our departure point, that is, p is the integer part of $\overline{u}\frac{\Delta t}{\Delta x}$. Therefore, our departure point at t^n, $\tilde{x}_j^n$, will lie in the interval $x_{i-p-1}^n \leq \tilde{x}_i^n < x_{i-p}^n$, as shown in Fig. 10.1.

Note: Given the magnitude of the advection winds, the departure point could be in a region that is several grid points downwind of the arrival grid point. This is an important feature to note for programming a semi-Lagrangian-based method.

The next step in the implementation of the semi-Lagrangian method is to calculate the distance from the departure point to the nearest grid point downwind of the departure point. This distance is quite often referred to as $a \equiv x_{i-p} - \tilde{x}_i^n$. Another way to consider this distance is as a correction term in a Taylor series expansion of $\psi\left(x_i^p\right)$, which is

$$\psi_i^{n+1} = \psi\left(x_i, t^{n+1}\right) = \psi\left(\tilde{x}_{i-p}, t^n\right) = \psi\left(x_{i-p} - \alpha\Delta x, t^n\right),$$

where $\alpha \equiv a\frac{\Delta t}{\Delta x}$ and $\alpha \in \left[0, \frac{1}{2}\right]$.

From this information it is possible to calculate the weighting, α, necessary for the interpolation, where α is defined as

$$\alpha = \frac{\left(x_{i-p} - \tilde{x}_i^n\right)\Delta t}{\Delta x}. \tag{10.5}$$

Now comes the important question: Which order of accuracy for the interpolation should be used? To start lets consider a linear interpolation. Thus the tracer's value at the departure point is

$$\psi\left(\tilde{x}_{i-p}, t^n\right) = \alpha\psi_{i-p-1}^n - (1-\alpha)\,\psi_{i-p}^n. \tag{10.6}$$

However, it can be shown quite easily when implemented into a computer code, that the linear interpolation semi-Lagrangian approach has a strong damping effect and so is not shape conserving, which is a desired property for an advection scheme.

Now let us consider a quadratic interpolation-based approach. As we see above for the linear interpolation approach, we use the values of ψ at the grid points either side of the departure point, weighted by their distance from the departure point. For the quadratic approach we consider an interpolation that is upwind of the departure point; we shall still use the points either side but now we include the value for ψ at x_{i-p+1}. The quadratic interpolation semi-Lagrange polynomial is given by

$$\psi\left(\tilde{x}_{i-p}, t^n\right) = \frac{\alpha(\alpha+1)}{2}\psi_{i-p-1}^n - \left(1-\alpha^2\right)\psi_{i-p}^n + \frac{\alpha(\alpha-1)}{2}\psi_{i-p+1}^n. \tag{10.7}$$

As we shall show when we apply the quadratic scheme above to advection in the Eady model later in this chapter, the quadratic scheme has overshoots but does not damp as severely as the linear approach.

Now we move on to a cubic-based interpolation polynomial, which is given by

$$\begin{aligned}\tilde{\psi}\left(\tilde{x}_i^n, t^n\right) = &-\frac{\alpha\left(1-\alpha^2\right)}{6}\psi_{i-p-2}^n + \frac{\alpha(1+\alpha)(2-\alpha)}{2}\psi_{i-p-1}^n \\ &+ \frac{\left(1-\alpha^2\right)(2-\alpha)}{2}\psi_{i-p}^n - \frac{\alpha(1-\alpha)(2-\alpha)}{6}\psi_{i-p+1}^n.\end{aligned} \tag{10.8}$$

The cubic interpolation polynomial-based scheme presented in (10.8) does not have the damping effects that have been mentioned for the linear and quadratic interpolations polynomials presented above. There are still some problems with the cubic-based polynomial, but they are not as severe as some of the effects for the two lower-order schemes.

To recap, the grid points that are used for the interpolation to find the value of the tracer at the departure point are given in Fig. 10.2 and we have summarized the values for the three interpolation polynomials mentioned, at each grid point for reference in Table 10.1.

From an algorithmic point of view, the way to implement a semi-Lagrangian scheme is as follows:

Step 1: Determine the departure point at $t = n$, $\tilde{x}_i^n$, such that the tracer arrives at x_i^{n+1} through the backward trajectory. If the advection wind, $\overline{u}$, is constant with respect to time and spacial direction of the advection, then the backward calculation is simply $\tilde{x}_i^n = x_i^{n+1} - \overline{u}\Delta t$.

Step 2: Determine if the departure point coincides with a grid point, recalling that this grid point may be several spacial steps away from the arrival grid point, and derive the number of grid points behind the arrival point, which is through the integer component of $\overline{u}\frac{\Delta t}{\Delta x}$.

Step 3a: If the departure point is at a grid point, then the value of the tracer at this point ψ_{i-p}^n becomes the value for ψ_i^{n+1}.

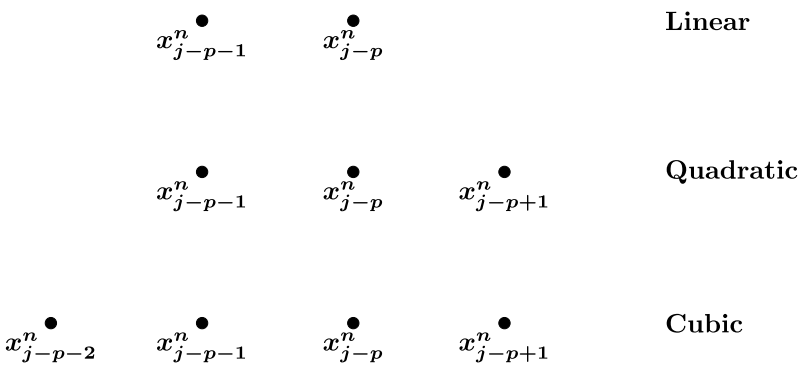


FIGURE 10.2

Visualization of the different points involved for the different order Lagrange interpolation polynomials.

Table 10.1 Summary of the Different Coefficients of the Lagrange Interpolation for the Linear, Quadratic, and Cubic-Based Semi-Lagrangian-Based Advection Schemes.

Order of Polynomial	$i-p-2$	$i-p-1$	$i-p$	$i-p+1$
Linear		α	$1-\alpha$	
Quadratic		$\frac{\alpha(\alpha+1)}{2}$	$(1-\alpha^2)$	$\frac{\alpha(\alpha-1)}{2}$
Cubic	$-\frac{\alpha(1-\alpha^2)}{6}$	$\frac{\alpha(1+\alpha)(2-\alpha)}{2}$	$\frac{(1-\alpha^2)(2-\alpha)}{2}$	$-\frac{\alpha(1-\alpha)(2-\alpha)}{6}$

Step 3b: If the departure point is not at a grid point, then determine the nearest grid point that is upwind of the departure point, x_{i-p}, and calculate the distance from the departure point to the nearest upwind grid point, $a = x_{i-p} - \tilde{x}_i^n$.

Step 4: Calculate the interpolation weight, α, as $\alpha = \frac{a\Delta t}{\Delta x}$.

Step 5: Select the order of the Lagrange interpolation polynomial: linear, quadratic, cubic, quartic, quintic, ..., or a different type of interpolation.

Step 6: Evaluate the interpolation and this becomes the value of the tracer at the arrival point, $\psi_i^{n+1} = f\left(\alpha, \psi^n_{i-p+\{1,2,\ldots,l\}}, \psi^n_{i-p-\{0,1,2,\ldots,k\}}\right)$. Note that l and k may not be equal.

Step 7: Return to Step 1 and evaluate for the next time level.

The three interpolation polynomials that have been presented in this section are the linear, quadratic, and cubic **Lagrange polynomials**. There are many different choices for interpolation polynomials and we shall consider different choices in the next section, where we start with the derivation of the linear, quadratic, and cubic Lagrange polynomials.

10.3 Interpolation Polynomials

In this section we shall consider different order polynomials for the Lagrange, Newton-divided difference, Hermite, and spline-based interpolation polynomials. Before we introduce the interpolation techniques, we briefly define some properties that we would like our interpolation polynomials to possess.

The first property that would be extremely useful for the interpolation polynomials to possess is **monotonicity**. We first define what a monotonic function is and then how this definition applies to interpolation polynomials.

Definition 10.1. A function f that is defined on a subset, S, of the real numbers, $\mathbb{R}$, is said to be **monotonic** if and only if it is entirely increase or decreasing over the subset. The function is referred to as **monotonically increasing** if for all the x_i and x_j that are in the subset, such that $x_i \leq x_j$ then $f(x_i) \leq f\left(x_j\right)$, and as such the function f preserves the ordering of the values in the set S.

If we reverse the inequality above, then we have a definition for **monotonically decreasing**. By association, if we were to tighten the inequalities to greater than or less than for the decreasing case, we would have the definition for **strictly monotonically increasing (decreasing)**.

This is an important property that we would wish that the interpolation polynomial that we are applying preserves, and that is the essence of the definition for a monotonic interpolation polynomial.

Definition 10.2. An interpolation polynomial is said to be a **monotonic interpolation** if it preserves the monotonicity of the underlying data.

As an example, linear interpolation preserves monotonicity, but not all higher-order interpolation polynomials do. What does it mean if a interpolation polynomial is not monotonic? In short, it means that while the polynomial will fit through the data, it may do what is referred to as **overshoot** or **undershoot** between points or features that are being advected.

10.3.1 Lagrange Interpolation Polynomials

The Lagrange interpolation polynomials play a vital part in some application of the semi-Lagrangian schemes. These sets of interpolation polynomial are named after Joseph Louis Lagrange, who published them in 1795. However, they had first been discovered in 1779 by Edward Wary and then rediscovered in 1783 by Leonhard Euler. We therefore have the following definition.

Definition 10.3. The Lagrange interpolating polynomial is the polynomial $P(x)$ of degree less than or equal to $(n-1)$ that passes through n points $(x_1, y_1 = f(x_1)), \ldots, (x_n, y_n = f(x_n))$ is given by

$$P(x) = \sum_{i=1}^{n} P_i(x), \quad \text{where } P_i(x) \equiv y_i \prod_{\substack{k=1 \\ k \neq i}}^{n} \frac{x - x_k}{x_i - x_k}. \tag{10.9}$$

We shall now apply the definition in (10.9) to derive the coefficients for the linear, quadratic, and cubic Lagrange interpolation polynomials that were stated in the last section. Note that we shall drop the i component from the subscripts to derive the three polynomials set around the point from the arrival point index.

The first step in deriving the coefficients for the three different orders of polynomial is to identify the expressions for the distances from the departure point and the surrounding grid points in terms of α. We have defined α to be the distance from the nearest upwind grid point to the departure point (i.e., $\alpha = x_p - \tilde{x}_i$). Therefore, if we consider the grid points either side of the departure point, and recall that the distance between each grid point has been normalized with respect to Δx, we have that $\tilde{x}_i - x_p = -\alpha$, $\tilde{x}_i - x_{p-1} = 1 - \alpha$, which occurs as a result of considering that the distance between the x_{p-1} grid point to the x_p grid point is 1, and then we have to go back α to arrive at the departure point. Following this same argument implies that the distance for $\tilde{x}_i$ to the x_{p-2} grid point is $2 - \alpha$.

Finally considering the distance in the upwind direction from the departure point we have that the distance $\tilde{x}_i - x_{p+1}$ is as follows; first, we have a $-\alpha$ from the departure point to the x_p grid point and then a further -1 between the x_p and x_{p+1} grid points. Therefore, we have that $\tilde{x}_i - x_{p+1} = -\alpha - 1 \equiv -(1+\alpha)$.

Given all these distances, it is now possible to derive the coefficients for the linear, $P_1(\alpha)$, quadratic, $P_2(\alpha)$, and cubic, $P_3(\alpha)$, Lagrange interpolation polynomials are as follows:

Linear: $\frac{(\tilde{x}_i^n - x_p)}{(x_{p-1} - x_p)}\phi_{p-1} + \frac{(\tilde{x}_j^n - x_{p-1})}{(x_p - x_{p-1})}\phi_p$, which is equivalent to $\frac{-\alpha}{-1}\psi_{p-1} + \frac{(1-\alpha)}{1}\psi_p$, which results in $P_1(\alpha) \equiv \alpha\psi_p + (1-p)\psi_{p-1}$.

Quadratic: To fit a quadratic polynomial through the departure point we require three points x_{p-1}, x_p, and x_{p+1}, which upon substituting these points into (10.9) results in

$$P_2(\alpha) = \frac{(\tilde{x} - x_p)(\tilde{x} - x_{p+1})}{(x_{p-1} - x_p)(x_{p-1} - x_{p+1})}\psi_{p-1} + \frac{(\tilde{x} - x_{p-1})(\tilde{x} - x_{p+1})}{(x_p - x_{p-1})(x_p - x_{p+1})}\psi_p + \frac{(\tilde{x} - x_{p-1})(\tilde{x} - x_p)}{(x_{p+1} - x_{p-1})(x_{p+1} - x_p)}\psi_{p+1}, \tag{10.10}$$

which is equivalent to

$$P_2(\alpha) = \frac{(\alpha^2 + \alpha)}{2}\psi_{p-1} + \left(\alpha^2 - 1\right)\psi_p + \frac{(\alpha^2 - \alpha)}{2}\psi_{p+1}. \tag{10.11}$$

Cubic: To be able to fit a cubic Lagrange polynomial then we require four points, which have already been identified: where we have the two points downwind of the departure point, x_{p-2} and x_{p-1}, and then the two grid points that are upwind x_p and x_{p+1}. Therefore the four coefficients of the cubic Lagrange polynomial $P_3(\alpha)$ are

$$\begin{aligned} C_1(\alpha) &= \frac{(\tilde{x}-x_{p-1})(\tilde{x}-x_p)(\tilde{x}-x_{p+1})}{(x_{p-2}-x_{p-1})(x_{p-2}-x_p)(x_{p-2}-x_{p+1})}, \\ &= \frac{(1-\alpha)(-\alpha)(-\alpha-1)}{(-1)(-2)(-3)}, \\ &= -\frac{\alpha(\alpha^2-1)}{6}. \end{aligned} \tag{10.12a}$$

$$\begin{aligned} C_2(\alpha) &= \frac{(\tilde{x}-x_{p-2})(\tilde{x}-x_p)(\tilde{x}-x_{p+1})}{(x_{p-1}-x_{p-2})(x_{p-1}-x_p)(x_{p-1}-x_{p+1})}, \\ &= \frac{(2-\alpha)(-\alpha)(-\alpha-1)}{(1)(-1)(-2)}, \\ &= \frac{\alpha(2-\alpha)(1+\alpha)}{2}. \end{aligned} \tag{10.12b}$$

$$\begin{aligned} C_3(\alpha) &= \frac{(\tilde{x}-x_{p-2})(\tilde{x}-x_{p-1})(\tilde{x}-x_{p+1})}{(x_p-x_{p-2})(x_p-x_{p-1})(x_p-x_{p+1})}, \\ &= \frac{(2-\alpha)(1-\alpha)(-\alpha-1)}{(2)(1)(-1)}, \\ &= \frac{(2-\alpha)(1-\alpha)(1+\alpha)}{2}. \end{aligned} \tag{10.12c}$$

$$\begin{aligned} C_4(\alpha) &= \frac{(\tilde{x}-x_{p-2})(\tilde{x}-x_{p-1})(\tilde{x}-x_p)}{(x_{p+1}-x_{p-2})(x_{p+1}-x_{p-1})(x_{p+1}-x_p)}, \\ &= \frac{(2-\alpha)(1-\alpha)(-\alpha)}{(3)(2)(1)}, \\ &= -\frac{\alpha(2-\alpha)(1+\alpha)}{6}. \end{aligned} \tag{10.12d}$$

Combining (10.12a)–(10.12d) with their respective tracer values at their grid points results in the cubic Lagrange interpolation polynomial as $P_3(\alpha) = C_1(\alpha)\psi_{p-2} + C_2(\alpha)\psi_{p-1} + C_3(\alpha)\psi_p + C_4(\alpha)\psi_{p+1}$, where upon substituting the equation in for the C_is results in the expression for the explicit cubic interpolation scheme stated in (10.8).

Exercise 10.4. *Derive the distances for $\tilde{x}_i - x_{p-3}$, $\tilde{x}_i - x_{p+2}$, and $\tilde{x}_i - x_{p+3}$ and use these distances to determine the quartic Lagrange interpolation polynomial, $P_4(\alpha)$ for a central, upwind, and downwind-based framework.*

10.3.2 Newton Divided Difference Polynomials

The basis of the Newton polynomial comes from a desire to write the n-order polynomial p_n in terms of a p_{n-1} polynomial plus correction terms,

$$p_n(x) = p_{n-1}(x) + C(x). \tag{10.13}$$

In general $C(x)$ is a polynomial of degree n. For the function, $f(x)$, the correction term becomes

$$C(x_i) = p_n(x_i) - p_{n-1}(x_i) = f(x_i) - f(x_i) = 0, \quad i = 0, \ldots, n-1. \tag{10.14}$$

Therefore, we have

$$C(x) = a_n (x - x_0) \cdots (x - x_{n-1}). \tag{10.15}$$

Given that $p_n(x_n) = f(x_n)$, we can obtain an expression for the a_ns as

$$a_n = \frac{f(x_n) - p_{n-1}(x_n)}{(x_n - x_0) \cdots (x_n - x_{n-1})}.$$

The coefficient a_n is called the nth-order Newton divided difference and when we have the function $f(x)$ it is defined as

$$a_n \equiv f(x_0, x_1, \ldots, n_n).$$

This then makes the polynomial

$$p(x_n) = p_{n-1}(x_n) + (x - x_0)(x - x_1) \cdots (x - x_{n-1}) f(x_0, x_1, \ldots, x_n). \tag{10.16}$$

The next stage in the derivation of the Newton divided difference polynomials is to consider the Lagrange formulation for the polynomial. We start by defining the polynomial $\Phi(x)$ as

$$\Phi(x) \equiv (x - x_0)(x - x_1) \cdots (x - x_{n-1}), \tag{10.17}$$

then we have

$$\Phi_n'(x_i) \equiv (x_i - x_0)(x_i - x_1) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_{n-1}). \tag{10.18}$$

The two polynomials in (10.17) and (10.18) should look quite familiar as they are Lagrange polynomial approximations. Therefore, substituting (10.17) and (10.18) into (10.13) we obtain an expression for the polynomial as

$$p_n(x) = \sum_{j=0}^{n} \frac{\Phi_n(x)}{(x - x_j)\Phi_n'(x_j)} f(x_j). \tag{10.19}$$

The expression in (10.19) is for the case where x is not at what we refer to as grid points, but it is sometimes referred to as nodes [17].

We now return to addressing the factors a_n, which are the coefficients of x^n in the polynomial $p_n(x)$. To obtain expressions for these coefficients we use the Lagrange formula at each nth-degree term in (10.19), such that we have

$$f[x_0, x_1, \ldots, x_n] = \sum_{j=0}^{n} \frac{f(x_j)}{\Phi_n'(x_j)}, \tag{10.20}$$

where

$$f[x_0, x_1, \ldots, x_n] = \frac{f[x_1, \ldots, x_n] - f[x_0, \ldots, x_{n-1}]}{x_n - x_0}, \tag{10.21}$$

and it is because of (10.21) that the name associated with this interpolation formula is the **divided difference**. It is possible to write (10.21) in term of the sum of two $n-1$-order polynomials as

$$p_n(x) = \frac{(x_n - x)\, p_{n-1}^{\{0:n-1\}}(x) + (x - x_0)\, p_{n-1}^{\{1:n\}}(x)}{x_n - x_0}, \tag{10.22}$$

where the polynomial $p_{n-1}^{\{0:n-1\}}(x)$ represents the polynomial of degree $n-1$ that interpolates $f(x)$ from the set of grid points $\{x_0, \ldots, x_{n-1}\}$ and $p_{n-1}^{\{1:n\}}$ represents the another polynomial of degree $n-1$ that interpolates $f(x)$ again but now from the set of grid points $\{x_1, \ldots, x_n\}$. Therefore, the terms in the formula given by (10.19) combined with (10.20) and (10.21), applied to our even-spaced numerical mesh, componentwise are

$$\begin{aligned}
p_0(x) &= f(x_0), \\
p_1(x) &= f(x_0) + (x - x_0) f[x_0, x_1], \\
&\equiv p_0(x) + (x - x_0) \frac{f(x_1) - f(x_0)}{(x_1 - x_0)}, \\
&\equiv p_0(x) + (x - x_0) \frac{f(x_1) - f(x_0)}{\Delta x}, \\
&\approx p_0(x) + (x - x_0) f'(x), \\
p_2(x) &= f(x_0) + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2], \\
&\equiv p_0(x) + (x - x_0) \frac{f(x_1) - f(x_0)}{(x_1 - x_0)} + (x - x_0)(x - x_1) \frac{\frac{f(x_1)-f(x_0)}{(x_1-x_0)} - \frac{f(x_2)-f(x_1)}{(x_2-x_1)}}{(x_2 - x_0)}, \\
&\equiv p_1(x) + (x - x_0)(x - x_1) \frac{\frac{f(x_1)-f(x_0)}{\Delta x} - \frac{f(x_2)-f(x_1)}{\Delta x}}{2\Delta x}, \\
&\approx p_1(x) + (x - x_0)(x - x_1) \frac{f'(x_1) - f'(x_0)}{2\Delta x}, \\
&\approx p_0 + (x - x_0) f'(x_0) + \frac{(x - x_0)(x - x_1)}{2} f''(x_0), \\
&\vdots \\
p_n(x) &= f(x_0) + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2] + \cdots \\
&\quad + (x - x_0)(x - x_1) \cdots (x - x_{n-1}) f[x_0, x_1, \ldots, x_n].
\end{aligned} \tag{10.23}$$

We can see from the derivation of the different order polynomials in (10.23) that these polynomials appear to utilize centered finite difference approximations to the different derivatives in a Taylor series expansion about the point x_0. The formula in (10.23) is referred to as the **Newton divide difference formula for the interpolation polynomial**.

At the moment the formula in (10.23) only approximates the value of the function $f(x)$ at a grid point. In the semi-Lagrangian advection approximation we know that our departure point is not at a grid point, otherwise we would not need to interpolate. Thus we need to consider the formulation of the Newton divide difference when $x \neq x_i$. If we now consider the point $\tilde{x}$ say, which is our departure

point, then the polynomial approximation of our function at $x = \tilde{x}$ is

$$p_{n+1}(x) = f(x_0) + (x - x_0) f[x_0, x_1] + \cdots + \left(\prod_{i=0}^{n-1} (x - x_i)\right) f[x_0, x_1, \ldots, x_n] + \left(\prod_{i=0}^{n} (x - x_i)\right) f\left[x_0, x_1, \ldots, x_n, \tilde{x}\right], \tag{10.24}$$

where in (10.24) we have the polynomial approximation for the function $f(x)$ at the grid point x_0 plus the interpolation term to the point $\tilde{x}$. Therefore, since the polynomial $p_{n+1}(x)$ in (10.24) is now an approximation to $f(\tilde{x})$, which then implies that if we let $x = \tilde{x}$ in (10.24), then we have an expression for the interpolation polynomial for $f(\tilde{x})$ as

$$f(\tilde{x}) = p_n(\tilde{x}) + \left(\prod_{i=0}^{n} (\tilde{x} - x_i)\right) f\left[x_0, x_1, \ldots, x_n, \tilde{x}\right]. \tag{10.25}$$

We now apply (10.25) to the interpolation to the departure point situation for the semi-Lagrangian schemes. If we consider the linear polynomial first, then we have

$$\begin{aligned} p_N^{(1)}(\tilde{x}) &\equiv \psi_{i-p} + \left(\tilde{x} - x_{i-p}\right)\left(\psi_{i-p} - \psi_{i-p-1}\right), \\ &= \psi_{i-p} - \alpha\left(\psi_{i-p} - \psi_{i-p-1}\right), \\ &= (1 - \alpha)\psi_{i-p} + \alpha\psi_{i-p-1}. \end{aligned} \tag{10.26}$$

Thus, the linear approximation for the Newton polynomial in (10.26) is equivalent to the linear Lagrange polynomial.

We shall now see if the quadratic Lagrange polynomial and the Newton divided difference polynomial are also equivalent. We have to form the first-order upwind approximations to the first derivative as

$$f\left[x_{i-p-1} - x_{i-p}\right] \equiv \frac{\psi_{i-p} - \psi_{i-p-1}}{\left(x_{i-p} - x_{i-p-1}\right)}, \quad f\left[x_{i-p+1}, x_{i-p}\right] \equiv \frac{\psi_{i-p+1} - \psi_{i-p}}{\left(x_{i-p+1} - x_{i-p}\right)}.$$

The next step is to form the second-order divided differences between the two first-order divided difference above; this then results in

$$f\left[x_{i-p-1}, x_{i-p}, x_{i-p+1}\right] = \frac{\frac{\psi_{i-p+1} - \psi_{i-p}}{\left(x_{i-p+1} - x_{i-p}\right)} - \frac{\psi_{i-p} - \psi_{i-p-1}}{\left(x_{i-p} - x_{i-p-1}\right)}}{(x_2 - x_1)}. \tag{10.27}$$

Substituting the distances between the grid points above and multiplying by $\left(\tilde{x} - x_{i-p-1}\right)\left(\tilde{x} - x_{i-p+1}\right)$ expressed in terms of the rational component of the Courant number results in

$$a_2 \equiv (1 - \alpha)(-\alpha)\frac{\left(\psi_{i-p+1} - 2\psi_{i-p} + \psi_{i-p-1}\right)}{2}. \tag{10.28}$$

To obtain the quadratic Newton divided distance interpolation polynomial, we add (10.28) to (10.26). Thus we arrive at

$$p_N^{(2)}(\tilde{x}) = (1 - \alpha)\psi_{i-p} + \alpha\psi_{i-p-1} + (1 - \alpha)(-\alpha)\frac{\left(\psi_{i-p+1} - 2\psi_{i-p} + \psi_{i-p-1}\right)}{2}.$$

Collecting the different factors of ψ results in

$$p_N^{(2)}(\tilde{x}) = \frac{\alpha+\alpha^2}{2}\psi_{i-p-1} + \left(1-\alpha^2\right)\psi_{i-p} + \frac{\alpha^2-\alpha}{2}\psi_{i-p+1}. \tag{10.29}$$

As with the linear Newton divided difference polynomial, the quadratic Newton interpolation polynomial can be shown to be equivalent to the quadratic Lagrange interpolation polynomial. We now consider the cubic Newton divided difference interpolation polynomial. We only have to evaluate the divided differences between two different numerical approximations to the second-order derivatives where we are now introducing another interpolation grid point at x_{i-p-2}, which results in

$$\begin{aligned} &f\left[x_{i-p-2}, x_{i-p-1}, x_{i-p}, x_{i-p+1}\right] \\ &= \frac{\dfrac{\psi_{i-p+1}-\psi_{i-p}}{\Delta x} - \dfrac{\psi_{i-p}-\psi_{i-p-1}}{\Delta x}}{2\Delta x} - \frac{\dfrac{\psi_{i-p}-\psi_{i-p-1}}{\Delta x} - \dfrac{\psi_{i-p-1}-\psi_{i-p-2}}{\Delta x}}{2\Delta x}}{3\Delta x}. \end{aligned} \tag{10.30}$$

The expression in (10.30) can be simplified for our case because Δx is equivalent to 1 as we have normalized the distances between the grid points. However, expanding the divided differences in (10.30) and simplifying results in

$$\begin{aligned} &f\left[x_{i-p-2}, x_{i-p-1}, x_{i-p}, x_{i-p+1}\right] \\ &= \frac{\frac{\psi_{i-p+1}-2\psi_{i-p}+\psi_{i-p-1}}{2\Delta x^2} - \frac{\psi_{i-p}-2\psi_{i-p-1}+\psi_{i-p-2}}{2\Delta x^2}}{3\Delta x}, \\ &= \frac{1}{6\Delta x^3}\left(\psi_{i-p+1} - 3\psi_{i-p} + 3\psi_{i-p-1} - \psi_{i-p-2}\right), \\ &\equiv \frac{1}{6}\left(\psi_{i-p+1} - 3\psi_{i-p} + 3\psi_{i-p-1} - \psi_{i-p-2}\right). \end{aligned} \tag{10.31}$$

This then enables us to define the a_3 coefficient as

$$\begin{aligned} a_3 &= (-\alpha)(1-\alpha)(-\alpha-1)\frac{1}{6}\left(\psi_{i-p+1} - 3\psi_{i-p} + 3\psi_{i-p-1} - \psi_{i-p-2}\right), \\ &= \frac{\alpha\left(1-\alpha^2\right)}{6}\left(\psi_{i-p+1} - 3\psi_{i-p} + 3\psi_{i-p-1} - \psi_{i-p-2}\right). \end{aligned} \tag{10.32}$$

Thus the expression for the cubic Newton divided difference polynomial is (10.29) plus (10.32), which results in

$$\begin{aligned} p_N^{(3)}(\tilde{x}) = &\frac{\alpha+\alpha^2}{2}\psi_{i-p-1} + \left(1-\alpha^2\right)\psi_{i-p} + \frac{\alpha^2-\alpha}{2}\psi_{i-p+1} \\ &+ \frac{\alpha\left(1-\alpha^2\right)}{6}\left(\psi_{i-p+1} - 3\psi_{i-p} + 3\psi_{i-p-1} - \psi_{i-p-2}\right). \end{aligned} \tag{10.33}$$

Collecting the different factors of the four ψ grid point values culminates in

$$\begin{aligned} p_N^{(3)} = &-\frac{\alpha-\alpha^3}{6}\psi_{i-p-2} + \left(\alpha + \frac{\alpha^2}{2} - \frac{\alpha^3}{2}\right)\psi_{i-p-1} + \left(1 - \frac{\alpha}{2} - \frac{\alpha^2}{2} + \frac{\alpha^3}{2}\right)\psi_{i-p} \\ &+ \left(-\frac{\alpha}{3} + \frac{\alpha^2}{2} - \frac{\alpha^3}{6}\right)\psi_{i-p+1}. \end{aligned} \tag{10.34}$$

Once again we have that the Newton divided difference interpolation polynomial for this order of approximation is the same as that for the Lagrange interpolation. While so far the first three Newton polynomials match the Lagrange, after some multiplying out parenthesis the reason to present the Newton-based interpolation is because of its ease of coding; we are literally forming finite difference approximation to the Taylor series of f. It is stated in many different sources that from a computational point of view, the Newton method is easier to expand to include another data point to the original set.

To add another data point at the end of the current set of points is quite easy, and once the new data point is in the set we simply build up the new difference for that point, given all the divided differences from the original set. However, in comparison to the Lagrange interpolation, where we would have to code the next order Lagrange polynomial formula which would be quite lengthy, compared to a simple change to a do or for loop, and then add a extra row to an array.

Exercise 10.5. *Derive the fourth-order Newton divided difference polynomial for the semi-Lagrange situation and show that it is equivalent, depending on which five data points you pick, to the relevant quartic Lagrange polynomial.*

10.3.3 Hermite Interpolating Polynomials

As we saw in the derivations of the Lagrange and the Newton divided differences interpolation polynomial, we are only using the function's values as constraints for the path the interpolation takes. However, there are a couple of methods to obtain a higher order of accuracy from the interpolation formula. The first approach is to increase the number of points to interpolate through; however, with this approach we are using more and more points that are further away from the region where we wish to evaluate the interpolation.

The second approach to improve interpolation polynomial accuracy is to fit the polynomial to higher-order derivative values of the function at the points surrounding the interpolation point. This then constrains the interpolation polynomial to match both $f\left(x_j\right)$ and $f'\left(x_j\right)$ for $j = 0, 1, \ldots, N$. One such set of interpolating polynomials that match these two conditions are referred to as **Hermite interpolating polynomials** and we show their derivation below.

We start by stating the problem which is we require our interpolating polynomial $p\left(x\right)$ to satisfy the following conditions

$$p\left(x_j\right) = f_j \text{ and } p'\left(x_j\right) = f'_j, \quad \text{for } j = 1, 2, \ldots, N, \tag{10.35}$$

where the x_js are our grid points, and f_j and f'_j are the given values of our function and its first derivative at the grid points. The expression in (10.35) defined $2N$ constraints on the interpolation polynomial and as such the highest degree the resulting interpolating polynomial can take is $2N - 1$.

We now introduce the following notation to help shorten expressions in the derivation of the Hermite polynomial:

$$\chi_N\left(x\right) = \left(x - x_1\right)\left(x - x_2\right)\cdots\left(x - x_n\right), \tag{10.36a}$$

$$\begin{aligned} L_j\left(x\right) &= \frac{\left(x - x_1\right)\cdots\left(x - x_{j-1}\right)\left(x - x_{j+1}\right)\cdots\left(x - x_N\right)}{\left(x_j - x_1\right)\cdots\left(x_j - x_{j-1}\right)\left(x_j - x_{j+1}\right)\cdots\left(x_j - x_N\right)}, \\ &\equiv \frac{\chi_n\left(x\right)}{\left(x - x_j\right)\chi'_n\left(x_j\right)}, \end{aligned} \tag{10.36b}$$

$$\tilde{h}_j(x) = (x - x_j)\left[L_j(x)\right]^2, \tag{10.36c}$$

$$h_j(x) = \left(1 - 2L_j'(x_j)(x - x_j)\right)\left[L_j(x)\right]^2. \tag{10.36d}$$

Given the definitions above, we have two extra conditions that are for $j, k = 1, \ldots, N$, which are

$$h_j(x_k) = \tilde{h}_j(x_k) = 0, \quad 1 \le j, k \le N, \tag{10.37a}$$

$$h_j(x_k) = \tilde{h}_j'(x_j) = \begin{cases} 0 & j \neq k, \\ 1 & j = k. \end{cases} \tag{10.37b}$$

The interpolating polynomial that satisfies the conditions in (10.35) is defined as

$$H_N(x) = \sum_{j=1}^{N} y_j h_j(x) + \sum_{j=1}^{N} y_j' \tilde{h}_j(x). \tag{10.38}$$

Given the expressions above for the Hermite interpolation polynomial, the most frequently used order of the Hermite polynomial is the cubic. The definitions above imply that we require the values of the tracer at points either side of the departure point, as well as the derivative of the tracer at these points. Therefore, we have

$$p(x_{i-p-1}) \equiv \psi_{i-p-1}, \qquad p'(x_{i-p-1}) \equiv \psi'_{i-p-1},$$
$$p(x_{i-p}) \equiv \psi_{i-p}, \qquad p'(x_{i-p}) \equiv \psi'_{i-p}.$$

The next step is to form the Lagrange basis L_1 and L_2, where for semi-Lagrange advection this will be L_{i-p-1} and L_{i-p}, respectively. Thus we have

$$L_{i-p-1} = \frac{x - x_{i-p}}{x_{i-p-1} - x_{i-p}}, \qquad L_{i-p} = \frac{x - x_{i-p-1}}{x_{i-p} - x_{i-p-1}},$$
$$L'_{i-p-1} = \frac{1}{x_{i-p-1} - x_{i-p}}, \qquad L'_{i-p} = \frac{1}{x_{i-p} - x_{i-p-1}}.$$

Next, forming the h_j and the $\tilde{h}_j$ results in

$$h_{i-p-1} = \left(1 + \frac{2(x_{i-p-1} - x)}{x_{i-p} - x_{i-p-1}}\right)\left(\frac{x_{i-p} - x}{x_{i-p} - x_{i-p-1}}\right)^2,$$

$$h_{i-p} = \left(1 + \frac{2(x_{i-p} - x)}{x_{i-p} - x_{i-p-1}}\right)\left(\frac{x - x_{i-p-1}}{x_{i-p} - x_{i-p-1}}\right)^2,$$

$$\tilde{h}_{i-p-1} = \frac{(x - x_{i-p-1})(x_{i-p} - x_{i-p})^2}{(x_{i-p} - x_{i-p-1})^2},$$

$$\tilde{h}_{i-p} = -\frac{(x - x_{i-p-1})^2(x_{i-p} - x)}{(x_{i-p} - x_{i-p-1})^2}.$$

The final form of the cubic Hermite polynomial is given by

$$H_2(x) = h_{i-p-1}\psi_{i-p-1} + h_{i-p}\psi_{i-p} + \tilde{h}_{i-p-1}\psi'_{i-p-1} + \tilde{h}_{i-p}\psi'_{i-p}. \tag{10.39}$$

Substituting the distances from the grid points to the departure points, and the plus or minus one for the distance between the grid points themselves, results in

$$H_2(x) = \left(1 + 2\frac{1-\alpha}{1}\right)\left(\frac{\alpha}{1}\right)^2 \psi_{i-p-1} + \left(1 + 2\frac{\alpha}{1}\right)\left(\frac{1-\alpha}{1}\right)^2 \psi_{i-p} + \frac{(1-\alpha)\alpha^2}{1^2}\psi'_{i-p-1} - \frac{(1-\alpha)^2\alpha}{1^2}\psi'_{i-p}. \tag{10.40}$$

Expanding out the brackets in (10.40) enables us to arrive at the simpler expressions for the cubic Hermite interpolation polynomial

$$H_2(x) = \left(3\alpha^2 - 2\alpha^3\right)\psi_{i-p-1} + \left(1 - 3\alpha^2 + 2\alpha^3\right)\psi_{i-p} + \left(\alpha^2 - \alpha^3\right)\psi'_{i-p-1} - \left(\alpha - 2\alpha^2 + \alpha^3\right)\psi'_{i-p}. \tag{10.41}$$

However, it may be the case that the derivatives are not known at the grid points and as such we require techniques that could be applied to approximate them. At this point it could appear that we have lost the advantage for the number of points needed to fit the interpolant compared to the Lagrange polynomial and the Newton divided difference polynomial, while this may be true for certain derivative estimates; as mentioned in [399], because of the flexibility of the Hermite interpolations it is possible, as shown in [479], that the resulting semi-Lagrangian scheme conservers monotonicity.

It is possible to define many different approximations to the derivatives that are required for the Hermite interpolation polynomial, where these estimates of the derivatives could be linear or nonlinear, but they do need to be *centered* at the grid points. Following the notation from [399], we define the discrete gradient operator Δ_j which is acting on the interval $\left[x_j, x_{j+1}\right]$, as

$$\Delta_j = \frac{\psi_{j+1} - \psi_j}{x_{j+1} - x_j}, \tag{10.42}$$

so that it is then possible to define approximations to the derivative at the points x_{j-1} and x_j through different combinations of the discrete gradient at nearby points.

In [399] there are three schemes that are listed, where one is a second-order approximation to the gradients and is calculated as an arithmetic mean from considering the discrete gradient at the points $j-1$ and j (i.e., $\psi'_j \approx \frac{\Delta_{j-1}+\Delta_j}{2}$). This scheme can be shown to be second order. Two other schemes that are presented in [399] are fourth order and are from [189,348]; they are given by

$$\psi'_j \approx \frac{-\Delta_{j-2} + 7\Delta_{j-1} + 7\Delta_j - \Delta_{j+1}}{12},$$
$$\psi'_j \approx \frac{-3\Delta_{j-2} + 19\Delta_{j-1} + 19\Delta_j - 3\Delta_{j+1}}{32},$$

respectively.

10.3.4 Cubic Spline Interpolation Polynomials

The spline set of interpolation polynomials are part of a general class of interpolation methods that are referred to as *piecewise polynomial interpolations*. In addition to being used for the interpolation of a value of a tracer to a departure points in semi-Lagrangian modeling, piecewise polynomials, but especially the **spline** family, are used in a large number of disciplines: data fitting, numerical integration,

as well as differentiation, numerical solutions to integral and differential equations. However, for our use here we shall focus on the piecewise polynomial theory as applied to interpolation problems.

We begin by considering a piecewise polynomial function that we shall denote as $p(x)$, which has an associated grid such that

$$-\infty < x_0 < x_1 < \cdots < x_N < \infty,$$

where the points indicated above are the bounds on the region where the piecewise polynomials are defined on. The function $p(x)$ is a polynomial that is defined on each interval which is defined as

$$(-\infty, x_0], [x_0, x_1], \ldots, [x_N, \infty). \tag{10.43}$$

It should be noted that in most cases the two intervals that contain either ∞ or $-\infty$ are usually not included in the fitting of the piecewise polynomials.

The order of the piecewise polynomial $p(x)$ is determined by being one order of magnitude higher than the largest magnitude of the piecewise polynomials over the intervals of consideration. An assumption that is made about $p(x)$ is that it does not have to be continuous, nor does its first derivative, but it is quite common for the piecewise polynomial to be continuous.

There are two possible classifications in general for piecewise polynomials interpolation problems; the first is referred to as a local problem and the second is referred to as a global problem. For semi-Lagrangian advection we consider spline functions that are associated with global problems.

Definition 10.6. A **spline function** $S(x)$ is said to be of order m that is greater than or equal to one on the interval

$$a = x_0 < x_1 < x_2 < \cdots < x_{N-1} < x_N = b,$$

if it satisfies the following two properties:

1. $S(x)$ is a polynomial of degree that is less than m on each of the subintervals $[x_{j-1}, x_j]$.
2. The k derivative of the spline function, $S^{(k)}(x)$, is continuous on the full interval $[a, b]$ for all of the derivatives up to $m-2$.

An interesting feature to remember about the derivatives of the spline functions of order m is that it too is a spline function but of order $m-1$. An opposite but equivalent property of the spline functions is also true for the integral, which means that the anti-derivative of a spline function of $m-1$ is a spline function but of order m.

One of the most commonly used order of spline functions are referred to as the **cubic splines**, where m would be 4. There are many reasons why cubic splines are popular; first, they are smooth functions with which to fit to data, but importantly for interpolation purposes, they do not have an oscillatory behavior that is common for higher-order degree polynomials associated with interpolation.

The interpolation problem can be stated as we wish to find a cubic spline, $S_4(x)$, such that $S(x_i) = y_i$ for $i = 1, 2, \ldots, N$. The general form of the cubic spline is given by

$$S_4(x) \equiv a_j + b_j x + c_j x^2 + d_j x^3, \tag{10.44}$$

where we have $x_{j-1} \le x \le x_j$ for $j = 1, 2, \ldots, N$. The equation in (10.44) contains four unknowns for each spline, a_j, b_j, c_j, and d_j for a total of $4N$ unknowns over the whole interval, which we will require

conditions on the splines to enable us to determine these coefficients. The first two constraints/conditions for each spline are (1) that the spline must satisfy $S_4(x)$ and (2) the continuity of the kth derivative property state earlier.

The second constraint implies that the kth derivative must be equal from either direction at the point x_j for each of the splines at each grid point. This constraint results in $4N-2$ conditions, but we require $4N$ conditions, implying that there are two degrees of freedom in choosing the coefficients in (10.44). Therefore, to obtain a unique interpolation spline we should expect to impose two more constraints.

We now consider how to define the cubic splines. We start by denoting the second derivative of the cubic splines $S_4''(x)$ as Q_j for $j=1,2,\ldots,N$. As we have chosen the spline to be a cubic polynomial, we know that its second derivative is a linear polynomial on the interval $\left[x_{j-1}, x_j\right]$. This implies that

$$S''(x) = \frac{\left(x_{j+1}-x\right)Q_j + \left(x-x_j\right)Q_{j+1}}{\Delta x}, \quad j=0,1,\ldots,N-1, \tag{10.45}$$

where Δx is still our grid spacing. However, in general it could be that the nodes are not equally spaced and the denominator in (10.45) would be replaced with Δx_j, where $\Delta x_j \equiv x_{j+1}-x_j$.

Given the expression for the second derivative of the spline in (10.45), then this derivative is continuous on the whole interval $[x_0, x_N]$. Thus with this continuity property it is possible to integrate (10.45) twice to obtain

$$S(x) = \frac{\left(x_{j+1}-x\right)^3 Q_j + \left(x-x_j\right)^3 Q_{j+1}}{6\Delta x} + A\left(x_{j+1}-x\right) + B\left(x-x_j\right), \tag{10.46}$$

where A and B are arbitrary constants to be determined. To determine the two constants A and B, we use the condition that the spline must be equal to the known values of the function y at the two grid points x_j and x_{j+1}, which implies that $S_4\left(x_j\right) = y_j$ and $S_4\left(x_{j+1}\right) = y_{j+1}$. Substituting these conditions into (10.46) and rearranging results in the expressions for the two constants as

$$A = \frac{y_j}{\Delta x} - \frac{\Delta x\, Q_j}{6}, \quad B = \frac{y_{j+1}}{\Delta x} - \frac{\Delta x\, Q_{j+1}}{6}.$$

Given the expression above for the two constants, we obtain the following

$$\begin{aligned} S_4(x) = {} & \frac{\left(x_{j+1}-x\right)^3 Q_j + \left(x-x_j\right)^2 Q_{j+1}}{6\Delta x} + \frac{\left(x_{j+1}-x\right)y_j + \left(x-x_j\right)y_{j+1}}{\Delta x}, \\ & - \frac{\Delta x\left(\left(x_{j+1}-x\right)Q_i + \left(x-x_j\right)Q_{i+1}\right)}{6}, \end{aligned} \tag{10.47}$$

for the cubic splines, where $x_j \le x \le x_{j+1}$, for $0 \le j \le N-1$.

The expression for the cubic splines in (10.47) implies that the splines are continuous on the interval $[x_0, x_N]$ as well as satisfying the interpolating condition that the splines must be equal to the value of the function at the grid points.

We now focus our attention to determining the constants Q_j, for $j=0,1,\ldots,N$. We have the condition that the first derivatives of the splines, $S'(x)$, are required to be continuous at the interior grid points, $x_1,\ldots,x_{N-1}$. The continuity condition is expressed mathematically as

$$\lim_{x\to x_j^+} S'(x) = \lim_{x\to x_j^-} S'(x), \quad \text{for } j=1,2,\ldots,N-1, \tag{10.48}$$

where the + superscript is referring to taking the limit from above the point x_j, and the − subscript refers to taking the limit from below the grid point x_j.

Given the new condition on the splines from (10.48), we now consider the first derivative of the cubic spline over two adjacent intervals. The first interval is $[x_j, x_{j+1}]$, which gives us

$$S'(x) = -\frac{(x_{j+1}-x)^2 Q_j + (x-x_j)^2 Q_{j+1}}{2\Delta x} + \frac{y_{i+1}-y_i}{\Delta x} - \frac{(Q_{i+1}-Q_i)\Delta x}{6}, \tag{10.49}$$

while on the adjacent interval, $[x_{j-1}, x_j]$, we have

$$S'(x) = -\frac{(x_j-x)^2 Q_{j-1} + (x-x_{j-1})^2 Q_j}{2\Delta x} + \frac{y_i-y_{i-1}}{\Delta x} - \frac{(Q_i-Q_{i-1})\Delta x}{6}. \tag{10.50}$$

Note: While we are deriving the cubic spline for an equally spaced grid, we should remember that if it is the case that the data points are not equally spaced, then the Δx terms in (10.49) and (10.50) should be replaced with Δx_j and Δx_{j-1}, respectively, to ensure the correct weighting of the distances between the grid points.

Therefore, taking the limit as $x \to x_j$ from above in (10.49) and from below in (10.50) and setting the two equations to be equal at that point results in the first terms in both equations tending to zero, which leaves only the terms involving y and Q as

$$\frac{\Delta x_{j-1}}{6} Q_{j-1} + \frac{\Delta x_j + \Delta x_{j-1}}{3} Q_j + \frac{\Delta x_j}{6} Q_{j+1} = \frac{y_{j+1}-y_j}{\Delta x_j} - \frac{y_j - y_{j-i}}{\Delta x_{j-1}}, \tag{10.51}$$

for $j = 1, 2, \ldots, N-1$ and where we have included the subscripts on the Δx terms to give the full expression for a non-equally spaced grid situation.

The formula in (10.51) results in $N-1$ equations for $N-1$ unknowns in the interior of the interval. However, we still require conditions at the end points of the interval to capture the last two remaining degrees of freedom associated with fitting cubic splines.

The more commonly used condition to capture the two remising degrees of freedom is based upon requiring the cubic spline, or any other order of spline, to be equal not just to the function at x_0 and x_N, but also to its first derivative values at these grid points as well, that is to say,

$$S'(x_0) = y'(x_0) \text{ and } S'(x_N) = y'(x_N).$$

Substituting these conditions into (10.49), evaluated at $x_j = x_0$ and $x_j = x_N$, yield the two end conditions:

$$\frac{\Delta x_0}{3} Q_0 + \frac{\Delta x_0}{6} Q_1 = \frac{y_1 - y_0}{\Delta x_0} - y'_0, \tag{10.52a}$$

$$\frac{\Delta x_{N-1}}{6} Q_{N-1} + \frac{\Delta x_{N-1}}{3} Q_N = y'_n - \frac{y_N - y_{N-1}}{\Delta x_{N-1}}. \tag{10.52b}$$

Combining the two end conditions above with the conditions for the interior points of the interval enables the problem to find the Q_js as the inversion of a matrix-vector equation $\mathbf{AQ} = \hat{\mathbf{y}}$, where

$$\hat{\mathbf{y}} = \begin{pmatrix} \frac{y_1 - y_0}{\Delta x_0} - y_0' \\ \frac{y_2 - y_1}{\Delta x_1} - \frac{y_1 - y_0}{\Delta x_0} \\ \vdots \\ \frac{y_N - y_{N-1}}{\Delta x_{N-1}} - \frac{y_{N-1} - y_{N-2}}{\Delta x_{N-2}} \\ y_N' - \frac{y_N - y_{N-1}}{\Delta x_{N-1}} \end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_{N-1} \\ Q_N \end{pmatrix}, \tag{10.53}$$

$$\mathbf{A} = \begin{pmatrix} \frac{\Delta x_0}{3} & \frac{\Delta x_0}{6} & 0 & \cdots & \cdots & 0 \\ \frac{\Delta x_0}{6} & \frac{\Delta x_0 + \Delta x_1}{3} & \frac{\Delta x_1}{6} & 0 & \cdots & 0 \\ 0 & & \ddots & & & \vdots \\ \vdots & & & \ddots & & \vdots \\ 0 & & & \frac{\Delta x_{N-2}}{6} & \frac{\Delta x_{N-2} + \Delta x_{N-1}}{3} & \frac{\Delta x_{N-1}}{6} \\ 0 & \cdots & \cdots & \cdots & \frac{\Delta_{N-1}}{6} & \frac{\Delta x_{N-1}}{3} \end{pmatrix}.$$

The matrix $\mathbf{A}$ is tri-diagonal, symmetric, positive definite, and diagonally dominant, which implies that the matrix is invertible and so there exists a unique solution to the matrix-vector problem.

We now address how to apply the cubic splines to the interpolation required in the semi-Lagrangian set up. A good explanation of how to apply the cubic spline approach for a semi-Lagrangian approximation in a transport model can be found in [371] and we use that explanation here.

The starting point is the formula of the general interpolant, for both the cubic Lagrange and the cubic spline, which is given by

$$\begin{aligned} \psi_i^{n+1} = \alpha \psi_{i-p-1}^n + (1-\alpha)\, \psi_{i-p}^n + \frac{1}{6}\left(C^3 - C\right)(\Delta x)^2\, \psi_{i-p-1}^{n''} \\ + \frac{1}{6}\left((1-\alpha)^3 - (1-\alpha)\right)(\Delta x)^2\, \psi_{i-p}^{n''}, \end{aligned} \tag{10.54}$$

where $\psi_{i-p}^{n''}$ is a discrete approximation to the second derivative of ψ at the grid point $i - p$.

It is stated in [371] that the Lagrange polynomial and the cubic spline approaches differ in their definitions for the second derivative $\psi_{i-p}^{n''}$. For the Lagrange interpolation we have

$$\psi'' = \frac{\psi_{i-p-1} - 2\psi_{i-p} + \psi_{i-p+1}}{(\Delta x)^2}. \tag{10.55}$$

If we insert (10.55) into (10.54), we obtain the cubic Lagrange interpolation formula from earlier. This is left as an exercise to verify this relationship. We move on to the equivalent expression for the cubic spline interpolation approach.

As we mentioned above, the cubic spline interpolant is derived from requiring that the first derivative of the interpolant be differentiable at the grid points. Given this condition, it is shown in [347] that our

interpolation polynomial must satisfy

$$\frac{\psi''_{i-p-1}}{6}+\frac{2\psi''_{i-p}}{3}+\frac{\psi''_{i-p+1}}{6}=\frac{\psi_{i-p-1}-2\psi_{i-p}+\psi_{i-p+1}}{(\Delta x)^2}. \tag{10.56}$$

If we now form the sum $\frac{1}{6}\psi^{n+1}_{i-p-1}+\frac{2}{3}\psi^{n+1}_{i-p}+\frac{1}{6}\psi^{n+1}_{i-p+1}$ and evaluate the right-hand side of (10.54) at the relative grid points, then we obtain

$$\begin{aligned}\frac{1}{6}\psi^{n+1}_{i-p-1}&=\frac{1}{6}\left(\alpha\psi^{n}_{i-p-2}+(1-\alpha)\,\psi^{n}_{i-p-1}+\frac{1}{6}\left(\alpha^3-\alpha\right)(\Delta x)^2\,\psi^{n''}_{i-p-2}\right.\\&\quad\left.+\frac{1}{6}\left((1-\alpha)^3-(1-\alpha)\right)(\Delta x)^2\,\psi^{n''}_{i-p-1}\right),\end{aligned} \tag{10.57a}$$

$$\begin{aligned}\frac{2}{3}\psi^{n+1}_{i-p}&=\frac{2}{3}\left(\alpha\psi^{n}_{i-p-1}+(1-\alpha)\,\psi^{n}_{i-p}+\frac{1}{6}\left(\alpha^3-\alpha\right)(\Delta x)^2\,\psi^{n''}_{i-p-1}\right.\\&\quad\left.+\frac{1}{6}\left((1-\alpha)^3-(1-\alpha)\right)(\Delta x)^2\,\psi^{n''}_{i-p}\right),\end{aligned} \tag{10.57b}$$

$$\begin{aligned}\frac{1}{6}\psi^{n+1}_{i-p+1}&=\frac{1}{6}\left(\alpha\psi^{n}_{i-p}+(1-\alpha)\,\psi^{n}_{i-p+1}+\frac{1}{6}\left(\alpha^3-\alpha\right)(\Delta x)^2\,\psi^{n''}_{i-p}\right.\\&\quad\left.+\frac{1}{6}\left((1-\alpha)^3-(1-\alpha)\right)(\Delta x)^2\,\psi^{n''}_{i-p+1}\right).\end{aligned} \tag{10.57c}$$

Summing the three expressions in (10.57a)–(10.57c), we can see that we have expressions that are the equivalent to the left-hand side of (10.56). We therefore have

$$\left(\alpha^3-\alpha\right)(\Delta x)^2\left(\frac{\psi^{n''}_{i-p-2}}{6}+\frac{2\psi^{n''}_{i-p-1}}{3}+\frac{\psi^{n''}_{i-p}}{6}\right)\equiv\left(\alpha^3-\alpha\right)\left(\psi^{n}_{i-p-2}-2\psi^{n}_{i-p-1}+\psi^{n}_{i-p}\right)$$

and

$$\begin{aligned}&\frac{1}{6}\left((1-\alpha)^3-(1-\alpha)\right)\left((\Delta x)^2\left(\frac{\psi^{n''}_{i-p-1}}{6}+\frac{2\psi^{n''}_{i-p}}{3}+\frac{\psi^{n''}_{i-p+1}}{6}\right)\right)\\&\quad\equiv\frac{1}{6}\left((1-\alpha)^3-(1-\alpha)\right)\left(\psi^{n}_{i-p-1}-2\psi^{n}_{i-p}+\psi^{n}_{i-p+1}\right).\end{aligned}$$

Substituting the expressions above into (10.57a)–(10.57c) and factorizing enables us to write the interpolation polynomial for the cubic spline in the form

$$\frac{1}{6}\psi^{n+1}_{i-p-1}+\frac{2}{3}\psi^{n+1}_{i-p}+\frac{1}{6}\psi^{n+1}_{i-p+1}=c_{-2}\psi^{n}_{i-p-2}+c_{-1}\psi^{n}_{i-p-1}+c_0\psi^{n}_{i-p}+c_1\psi^{n}_{i-p+1}. \tag{10.58}$$

The equivalent cubic Lagrange interpolation polynomial of (10.58) is

$$\psi i^{n+1}=l_{-2}\psi^{n}_{i-p-2}+l_{-1}\psi^{n}_{i-p-1}+l_0\psi^{n}_{i-p}+l_1\psi^{n}_{i-p+1}. \tag{10.59}$$

Below we have put side by side the definitions of the coefficients of the cubic spline and the cubic Lagrange interpolation polynomials to illustrate the differences between the two approximations:

$$l_{-2}\equiv\frac{\alpha^3-\alpha}{6},\qquad\qquad c_{-2}\equiv\frac{\alpha^3}{6}, \tag{10.60a}$$

$$l_{-1} \equiv \frac{-\alpha^3 + \alpha^2 + 2\alpha}{2}, \qquad c_{-1} \equiv \alpha + \frac{(1-\alpha)^3 - 2\alpha^3}{6}, \tag{10.60b}$$

$$l_0 \equiv \frac{\alpha^3 - 2\alpha^2 - \alpha + 2}{2}, \qquad c_0 \equiv (1-\alpha) + \frac{\alpha^3 - 2(1-\alpha)^3}{6}, \tag{10.60c}$$

$$l_1 \equiv \frac{-\alpha^3 + 3\alpha^2 - 2\alpha}{6}, \qquad c_1 \equiv \frac{(1-\alpha)^3}{6}. \tag{10.60d}$$

It is quite clear from (10.60a)–(10.60d) that there are substantial differences between the coefficients for the cubic Lagrange, and cubic spline, interpolation polynomials. There is also an important difference on the left-hand side of the interpolation polynomials. For the cubic spline interpolation polynomial, we have an implicit expression. As stated in [371], we have to solve a matrix equation for the cubic spline approach to obtain the values of the tracer at the departure point in the form $\mathbf{M}\boldsymbol{\psi}^{n+1} = \mathbf{S}\boldsymbol{\psi}^n$. Thus if we apply a cubic spline approach, we have to invert $\mathbf{M}$ to obtain an approximation to $\psi(\tilde{x}_i)$.

Exercise 10.7. *Verify that the terms for the Lagrange interpolation formula in* (10.60a)–(10.60d)*, when substituted into* (10.54)*, result in the equation for the cubic Lagrange interpolation formula.*

10.3.5 Shape-Conserving Semi-Lagrangian Advection

We have seen that we would like our interpolation scheme for finding the departure point to ensure that the associated advection remains monotonic if it is not diffussive. However, there are many other properties that are desired for the semi-Lagrangian scheme. One of these is *shape conserving*. In [479] such a scheme is presented.

We summarize here the derivation of shape-conserving schemes from [479]. If we consider the one-dimensional problem, with a grid of equally spaced points, $x_1 < x_2 < \cdots < x_N$, with associated data values at each point, given by $f_i = f(x_i)$, for $i = 1, 2, \ldots, N$, then we define a piecewise interpolant, p, on each of the grid intervals $[x_i, x_{i+1}]$ as

$$p(x) = p_i(\alpha),$$

where

$$p(x_i) = f_i,$$
$$\frac{dp}{dx}(x_i) = d_i,$$

for $i = 1, 2, \ldots, N$ and where d_i is some estimate of the derivative of the function f at the grid points. We should note that this is the definition of a Hermite polynomial. This then makes the interpolant a function of f_i and d_i at the end points of the intervals.

It is stated in [479] that it is possible to state two different forms of shape-conserving methods, which come from [92,189]. The shape-conserving methods can be expressed in a general form from [92], as

$$p_i \equiv \frac{P_i(\alpha)}{Q_i(\alpha)}, \tag{10.61}$$

on the interval where $0 \leq \alpha \leq 1$, which is equivalent to $x_i \leq x \leq x_{i+1}$. The two functions of the parameter α in (10.61) are defined as

$$P_i(\alpha) = f_{i+1}\alpha^3 + \left(r_i f_{i+1} - \Delta x_i d_{i+1}\right)\alpha^2(1-\alpha) + \left(r_i f_i + \Delta x_i d_i\right)\alpha(1-\alpha)^2 + f_i(1-\alpha)^3, \tag{10.62a}$$

$$Q_i(\alpha) = 1 + (r_i - 3)\alpha(1-\alpha). \tag{10.62b}$$

While the expressions above are a convenient way of expressing the interpolation polynomial, we should note that this may not be the most efficient way of applying them in practice.

If we substitute $r_i = 3$ into (10.62a) and (10.62b), then $Q_i = 1$, $\forall i$, and the $P_i(\alpha)$ would become the general form of the cubic spline. The second shape-conserving scheme from [92], which is referred to as the rational cubic interpolant, sets r_i to be

$$r_i = 1 + \frac{c_{i+1}}{c_i} + \frac{c_i}{c_{i+1}}, \tag{10.63}$$

where the c_is are differences between the discrete slope operator Δ and the continuous derivative as

$$c_i = \Delta_i - d_i,$$
$$c_{i+1} = d_{i+1} - \Delta_i,$$

and as a reminder the discrete slope is given by

$$\Delta_i = \frac{\left(f_{i+1} - f_i\right)}{\left(x_{i+1} - x_i\right)}.$$

In the companion paper to [479], [480], the authors refer to two different constraints for preserving the shape of the field that is being advected. These two possible approaches are through imposing a monotonic constraint or secondly a convex/concave constraint. We shall consider the two constraints in this order, starting with the monotonic-based constraint.

Monotonic-based shape-conserving constraints

For different interpolants to maintain monotonicity, the estimates of the derivative, d_i, must satisfy certain conditions. These conditions are enforced through constraints of the derivative estimates d_i and d_{i+1} at the end points of the grid interval as a function of the discrete slope, Δ_i, across the interval. However, we should note that the constraint on the derivative estimate d_i based upon the discrete slope Δ_i to the left may be different from the constraint that is based upon the discrete slope Δ_i of the interval to the right.

Given the possible differences in the discrete slopes from the two different intervals, we have some choices to make. The first approach would be to constrain the derivatives differently for the interpolation over the two intervals; in doing so the resulting interpolant would be C^0 continuous, which as noted the optimal control chapter, Chapter 7, means that the interpolant is continuous but its first derivative contains jumps. A second approach would be to enforce that the constraints from both intervals be satisfied simultaneously; this would then make the interpolant C^1 continuous, which implies that both the function and its first derivative are continuous.

In [479] there are two conditions that are defined to ensure that the interpolant satisfies one of the two continuity conditions just mentioned. The first condition is referred to as **necessary condition for monotonicity $\mathbf{C^0}$ (NCM0)**. The NCM0 condition is that for the interpolating function to be monotonic

in the grid interval, the derivative estimate at the end point must have the same sign as the discrete slope on the interval:

$$\begin{aligned} &\text{sign}\,(d_i) = \text{sign}\,(\Delta_i) = \text{sign}\left(\Delta_{i+1}\right), && \Delta_i \neq 0, \\ &d_i = d_{i+1} = 0, && \Delta_i = 0. \end{aligned} \tag{10.64}$$

The second condition stated in [479] is referred to as the **necessary condition for monotonicity $\mathbf{C^1}$ NCM1**. For the interpolating function to be continuous between intervals, the derivative estimates at a grid point must have the same sign as the discrete slope of the adjacent grid intervals:

$$\begin{aligned} &\text{sign}\left(\Delta_{i-1}\right) = \text{sign}\,(d_i) = \text{sign}\,(\Delta_i), && \Delta_{i-1}\Delta_i > 0, \\ &d_i = 0, && \Delta_{i-1}\Delta_i \leq 0. \end{aligned} \tag{10.65}$$

The reason for the choice of r in [92], that is mentioned in [479], is to ensure that the conditions NCM0 and NCM1 were also sufficient conditions for monotonicity for their resulting rational cubic interpolant. We should note here, again as stated in [479], that the two conditions, NCM0 and NCM1, are only necessary conditions for the cubic Hermite interpolant.

However, it is possible to arrive at a set of necessary and sufficient condition for monotonicity of the cubic Hermite interpolant as derived in [146]. This necessary and sufficient condition is stated as follows: if $\Delta_i = 0$, $d_i = d_{i+1} = 0$ then the necessary conditions (10.64) and (10.65) are also sufficient.

If the discrete slope Δ_i is not equal to zero, then we first define the ratio of the derivative estimates to the discrete slope by

$$\tau = \frac{d_i}{\Delta_i}, \tag{10.66a}$$

$$\eta = \frac{d_{i+1}}{\Delta_i}, \tag{10.66b}$$

then for $\Delta_i \neq 0$, the Hermite cubic interpolant will be monotonic if and only if

$$0 \leq \tau \leq 3, \qquad 0 \leq \eta \leq 3, \tag{10.67}$$

or

$$(\tau - 1)^2 + (\tau - 1)(\eta - 1) + (\eta - 1)^2 - 3(\tau + \eta + 2) \leq 0. \tag{10.68}$$

The constraints above may appear a bit random that the ratio of the derivative values to the discrete gradient slope at the various grid points need to be constrained by 3. Therefore, as a quick aside we shall briefly summarize the derivation of the constraints in (10.67) and (10.68) from [146].

Aside to Fritsch and Carlson [146]

We start by considering the general definition for a cubic polynomial over an interval $I_i = [x_{i-1}, x_i]$, where the point that we wish to interpolate to x is in this interval. Bear in mind the motivation that we have been using so far in this chapter, which is to construct a piecewise cubic function $p(x)$ that is $C^1[\mathbf{I}]$, where $\mathbf{I} \equiv [x_1, x_N]$, such that the polynomial is equal to the value of the fields at the grid points.

On each subinterval $I_i = [x_{i-1}, x_i]$ we define a cubic Hermite-based polynomial as

$$p(x) \equiv f_{i-1} H_1(x) + f_i H_2(x) + d_{i-1} H_3(x) + d_i H_4(x), \tag{10.69}$$

where here the d_js where $j = i-1, i$ are the derivative of the polynomial at the relative grid points, and the $H_k(x)$ are the Hermite basis functions for the specific interval defined by,

$$\begin{aligned}
H_1 &\equiv 3\left(\left(\frac{x_i - \tilde{x}}{\Delta x_{i-1}}\right)^2\right) - 2\left(\left(\frac{x_i - \tilde{x}}{\Delta x_{i-1}}\right)^3\right), \\
H_2 &\equiv 3\left(\left(\frac{\tilde{x} - x_{i-1}}{\Delta x_{i-1}}\right)^2\right) - 2\left(\left(\frac{\tilde{x} - x_{i-1}}{\Delta x_{i-1}}\right)^3\right), \\
H_3 &\equiv -\Delta x_i\left(\left(\frac{x_i - \tilde{x}}{\Delta x_{i-1}}\right)^3 - \left(\frac{x_i - \tilde{x}}{\Delta x_{i-1}}\right)^2\right), \\
H_4 &\equiv \Delta x_i\left(\left(\frac{\tilde{x} - x_{i-1}}{\Delta x_{i-1}}\right)^2 - \left(\frac{\tilde{x} - x_{i-1}}{\Delta x_{i-1}}\right)^2\right).
\end{aligned} \tag{10.70}$$

The goal of [146] is to derive conditions to ensure that the derivative approximations needed for the cubic Hermite-based approach are monotonic. Fritsch and Carlson's starting point is to state the monotonicity conditions: (10.66a) and (10.66b).

The next step is to assume that the discrete slope, Δ_{i-1}, which is equivalent to $\Delta_{i-1} = \frac{f_i - f_{i-1}}{\Delta x_{i-1}}$, is not equal to zero. Then we expand the polynomial in (10.69) with the definitions from (10.70) as a Taylor series expansion about $x = x_i$ which results in a cubic interpolation polynomial just in terms of $(\tilde{x} - x_{i-1})$, f_{i-1}, d_i, d_{i-1}, and Δ_{i-1} as

$$\begin{aligned}
p(\tilde{x}) = &\left(\frac{d_{i-1} + d_i - 2\Delta_{i-1}}{\Delta x_{i-1}}\right)(\tilde{x} - x_{i-1})^3 + \left(\frac{-2d_{i-1} - d_i + 3\Delta_{i-1}}{\Delta x_{i-1}}\right)(\tilde{x} - x_{i-1})^2 \\
&+ d_{i-1}(\tilde{x} - x_{i-1}) + f_{i-1}.
\end{aligned} \tag{10.71}$$

Exercise 10.8. *Verify the expression in* (10.71). ***Hint:*** *Expand* x_i *as* $x_i \equiv x_{i-1} + \Delta x_{i-1}$ *and* f_i *as* $f_i = f_{i-1} + \Delta_i \Delta x_{i-1}$.

Differentiating (10.71) with respect to $\tilde{x}$ results in

$$p'(\tilde{x}) = 3\left(\frac{d_{i-1} + d_i - 2\Delta_{i-1}}{\Delta x_{i-1}}\right)(\tilde{x} - x_{i-1})^2 + 2\left(\frac{-2d_{i-1} - d_i + 3\Delta_{i-1}}{\Delta x_{i-1}}\right)(\tilde{x} - x_{i-1}) + d_{i-1}. \tag{10.72}$$

Differentiating a second time to obtain the Hessian results in

$$p''(\tilde{x}) = 6\left(\frac{d_{i-1} + d_i - 2\Delta_{i-1}}{\Delta x_{i-1}}\right)(\tilde{x} - x_{i-1}) + 2\left(\frac{-2d_{i-1} - d_i + 3\Delta_{i-1}}{\Delta x_{i-1}}\right). \tag{10.73}$$

We now consider two cases that can arise:

Case I: If we have $d_{i-1} + d_i - 2\Delta_{i-1} = 0$, then for this situation the interpolant is either quadratic of linear, and therefore the associated derivative is linear or a constant, respectively. Since we have that $\min(d_{i-1}, d_i) \leq p(\tilde{x}) \leq \max(d_{i-1}, d_i)$, then the standard sign monotonicity condition is a sufficient condition.

Case II: If we have $d_{i-1} + d_i - 2\Delta_i \neq 0$, then the derivative of the interpolant is quadratic. The derivative will be concave up if $d_{i-1} + d_i - 2\Delta_{i-1} > 0$ and therefore concave down if $d_{i-1} + d_i - 2\Delta_{i-1} < 0$. In [146] they note that if $f_{i-1} < f_i$ and $p'(\tilde{x})$ is concave down then the interpolant

is monotone increasing, since $0 \leq \min(d_{i-1}, d_i) \leq p'(\tilde{x})$. By a similar argument, with the signs reversed in the inequalities above and the min replaced with the max operator, the interpolant is monotone decreasing.

Given that it is possible for the interpolant to be monotone increasing or decreasing, we need a condition that can accommodate both properties. Thus, let τ_{i-1} and κ_{i-1} be as defined in (10.66a) and (10.66b), which are the respective ratios of the endpoint derivatives to the slope of the secant line. This enables us to rewrite the condition for Case II as $(\tau_{i-1} + \kappa_{i-1} - 2)\,\Delta_i$ and therefore the interpolant is monotone if $\tau_{i-1} + \kappa_{i-1} - 2 < 0$, which is one of the conditions stated in [479].

If we now consider the other situation where $\tau_{i-1} + \kappa_{i-1} - 2 > 0$, then it is possible to show that the derivative of the interpolant has a unique extremum at

$$\hat{x} = x_{i-1} + \frac{\Delta x_i}{3}\left(\frac{2\tau_{i-1} + \kappa_{i-1} - 3}{\tau_{i-1} + \kappa_{i-1} - 2}\right), \tag{10.74}$$

and

$$p'(\hat{x}) = \Psi(\tau_{i-1}, \kappa_{i-1})\,\Delta_{i-1}, \tag{10.75}$$

where

$$\Psi(\tau, \kappa) = \tau - \frac{1}{3}\frac{(2\tau + \kappa - 3)^2}{(\tau + \kappa - 2)}. \tag{10.76}$$

Given these expressions above, we have arrived at the conditions that are stated in [479] for concave monotonicity.

Returning to Williamson and Rasch [479,480]

In [479] it is highlighted that the elliptical component of the necessary and sufficient condition in (10.68) is difficult to apply because the derivative estimates at the two ends of a grid interval may depend on each other. The simpler condition in (10.67), which they refer to as a rectangular condition, does provide a sufficient but not necessary condition to ensure monotonicity of the Hermite cubic interpolation polynomial [88]. The sufficient condition in (10.67) is obviously easier to enforce than the more general necessary and sufficient conditions from (10.68) and (10.67).

Given an estimate of the derivatives, which we should note are not guaranteed to satisfy any of the conditions for monotonicity that have been stated, if a monotonic interpolant is the desired outcome, then the derivative estimates must be modified to satisfy the appropriate condition. It is stated in [479] that a C^0 and C^1 form are defined on whether the derivative estimates d are bounded only by Δ on the interval being interpolated or by the discrete slopes of the two adjacent intervals simultaneously.

The modifications just mentioned are defined in [479] for both the cubic rational interpolant and for the cubic Hermite interpolant as follows:

Cubic rational: If the necessary conditions (10.66a) or (10.66b) are not met, depending if C^0 or C^1 continuity is the preferred outcome, then the appropriate derivative estimate is set to zero, as the sign of 0 can be considered either $+$ or $-$.

Cubic Hermite: If the derivative estimate does not satisfy (10.66a) or (10.66b), again depending of which form of continuity is sought, then we follow the procedure set out for the cubic rational. After

these conditions are not satisfied, if the conditions in (10.67) is not met, that is to say if $\tau \geq 3$ or $\eta \geq 3$, then these estimates are set to 3, which then gives both C^0 and C^1 continuity.

In Section 10.3.3 we introduced the Hyman derivative estimate, which is defined as

$$d_i = \frac{-f_{i+2} + 8f_{i+1} - 8f_{i-1} + f_{i-2}}{-x_{i+2} + 8x_{i+1} - 8x_{i-1} + x_{i-2}}, \tag{10.77}$$

which is a fourth-order estimate on a smooth varying grid. Another alternative is to consider a cubic derivative estimate, which arises from fitting a cubic Lagrange interpolant through the four points surrounding the interval. The derivatives at the end of the grid intervals are derived by differentiating the Lagrange estimate and evaluating it at the end of the grid interval.

It is important to note here that if the Lagrange estimate derivatives are not modified, then a Hermite interpolation polynomial with these estimates is equivalent to the cubic Lagrange interpolation polynomial, which is not monotonic. However, one of the pleasant properties of the Hermite interpolants is that it is possible to modify their derivative estimates to ensure monotonicity of the interpolants.

Another important thing to note is that depending on where the point is that we wish to interpolate our grid point values of the function to, with respect to the side if x_i in the interval (x_{i-1}, x_{i+1}), the associated four grid points needed for the cubic approximation to the derivative will either use x_{i-2}, x_{i-1}, x_i, x_{i+1} for $x \in (x_{i-1}, x_i)$ or x_{i-1}, x_i, x_{i+1}, x_{i+2} if $x \in (x_i, x_{i+1})$. Therefore, the estimates for d_i in these two subintervals are given by

$$d_i = \begin{cases} f\left[x_i, x_{i-1}\right] + \left(x_i - x_{i-1}\right) f\left[x_{i+1}, x_i, x_{i-1}\right] \\ \quad + \left(x_i - x_{i-1}\right)\left(x_i - x_{i+1}\right) f\left[x_{i+2}, x_{i+1}, x_i, x_{i-1}\right] & x \in \left(x_i, x_{i+1}\right), \\ f\left[x_{i_1}, x_{i-2}\right] + \left(\left(x_i - x_{i-1}\right) + \left(x_i - x_{i-2}\right)\right) f\left[x_i, x_{i-1}, x_{i-2}\right] \\ \quad + \left(x_i - x_{i-2}\right)\left(x_i - x_{i-1}\right) f\left[x_{i+1}, x_i, x_{i-1}, x_{i-2}\right] & x \in \left(x_{i-1}, x_i\right), \end{cases} \tag{10.78}$$

where the function $f[\cdot]$ is the Newton divided difference function that was defined in Section 10.3.2. Using the difference described in (10.78) is a piecewise interpolant will only enable the interpolant to be C^0 continuous.

A third approximation to the d_is that is considered in [479] is referred to as the Akima estimate [2], and it is the weighted average of the discrete slope on the two sides of the point that we wish to interpolate the function two, in our case for semi-Lagrangian advection the departure point. The Akima derivative estimate is given by

$$d_i = \begin{cases} \frac{\iota\Delta_{i-1} + \kappa\Delta_i}{\iota + \kappa}, & \iota + \kappa \neq 0, \\ \frac{\Delta_{i-1} + \Delta_i}{2}, & \iota + \kappa = 0, \end{cases} \tag{10.79}$$

where we have used ι and κ instead of ν and η, because their definitions are different, and we do not want to be confused about which definition of a variable we are using at this specific time. This is also why we have not used α and β, which are the coefficients in [479], because α has already been used in this chapter.

At first glance, (10.79) does not appear nonlinear; however, when we consider the definitions of ι and κ,

$$\begin{aligned} \iota &\equiv \left|\Delta_{i+1} - \Delta_i\right|, \\ \kappa &\equiv \left|\Delta_{i-1} - \Delta_{i-2}\right|, \end{aligned} \tag{10.80}$$

then we can see how (10.79) is nonlinear. The weights in (10.79), $\frac{\iota}{\iota+\kappa}$ and $\frac{\kappa}{\iota+\kappa}$ are inversely proportional to an approximation to the curvature on that side. The definition in (10.79) is often referred to as a *geometric mean* estimation [146].

As with the finite difference approaches, we have to consider whether or not there are restrictions on the spatial and temporal step sizes that ensure the stability of the semi-Lagrangian-based schemes.

10.4 Stability of Semi-Lagrangian Schemes

In this section we shall use the notation and expression from [399] which helps explain the difference from standard Von Neumann stability derivations, as the departure points are not always at the spatial grid points, and as such we have to account for that in the spacial analysis. The starting point for the stability analysis in [399] is to define the initial data for the advection equation as

$$\psi_0(x) = e^{ik\Delta x}, \tag{10.81}$$

where k is the wavenumber, and so the analytical solution to the advection equation is

$$u(x,t) = e^{ik(x-\overline{u}\Delta t)}. \tag{10.82}$$

If we now consider solutions that are separated by the time interval Δt, then they are related by

$$\begin{aligned} u(x,t+\Delta t) &= e^{ik(x-\overline{u}(t+\Delta t))}, \\ &= e^{ik(x-\overline{u}t-\overline{u}\Delta t)}, \\ &= e^{ik(x-\overline{u}t)}e^{-ik\overline{u}\Delta t}, \\ &= e^{-ik\overline{u}\Delta t}u(x,t). \end{aligned} \tag{10.83}$$

For a general initial data for the advection equation, the solutions at different times are related through the evolution operator, E, as

$$u(x,t+\Delta t) = \mathsf{E}(\Delta t)\,u(x,t) \equiv u(x-\overline{u}\Delta t, t), \tag{10.84}$$

where we have used the fact that the term involving Δt is independent of t and as such acts upon the x component. The factor $e^{-ik\overline{u}\Delta t}$ in (10.83) is referred to as the **Fourier symbol** of the evolution operator. Differences in the modulus and argument between the numerical and the analytic Fourier symbols represent errors in the amplitude and phase, respectively, of the numerically advected wave.

Given the derivation above, we know that the numerical solution depends upon four variables: the time step Δt, the spatial step size Δx, the wavenumber k, and the wave speed $\overline{u}$. We now introduce

two dimensionless parameters; the first is referred to as the **Courant number**, which is related to our semi-Lagrangian parameters as ν,

$$\nu = \overline{u}\frac{\Delta t}{\Delta x},$$

and indicates the number of spatial grid points that have been crossed by a trajectory in one time step.

The wavenumber of the analytical solution is related to the wavelength λ through

$$k = \frac{2\pi}{\lambda}. \tag{10.85}$$

If we consider the ratio of the wavelength to the mesh, $\lambda\Delta x$, then is possible to define a dimensionless wavenumber, ϕ as

$$\phi = \frac{2\pi}{\lambda\Delta x} \equiv k\Delta x. \tag{10.86}$$

Thus we can now define the analytical Fourier symbol, F, as

$$\begin{aligned} F = e^{-ik\overline{u}\Delta t} &= e^{-\phi\overline{u}\frac{\Delta t}{\Delta x}}, \\ &= e^{-\nu\phi}. \end{aligned} \tag{10.87}$$

We now determine the equivalent properties to those derived above for the numerical scheme. We start by introducing the spatial shift operator, S, which is defined as

$$\mathsf{S}\psi_p = \psi_{p+1}. \tag{10.88}$$

An important feature of the semi-Lagrangian schemes that we have to take note of here is that the finite difference stencil associated with the semi-Lagrangian schemes adapts to the trajectory of the flow; by this it is meant that for the Eulerian-based finite differences, we take the difference between points that are at the same location at every time step, which is not the case for the semi-Lagrangian-based advection schemes.

To represent the mechanism that the stencil for the semi-Lagrangian-based numerical schemes are flow dependent, we have to split the Courant number into its integer and non-integer components, where the integer component represents the nearest grid point and the non-integer component represents the distance from the nearest grid point:

$$\nu = \mathrm{p} + \alpha, \quad 0 \leq \beta < 1, \quad \mathrm{I} \in \{0, 1, 2, \ldots\}, \tag{10.89}$$

where the integer I represent the number of grid points behind or in front the nearest grid point at time $t = n$ to the departure point from ψ_i, and α is the ratio of the distance to the nearest upwind point of the grid space.

10.4.1 Stability Analysis of the Linear Lagrange Interpolation

We start the stability analysis for the different Lagrange interpolation with the linear Lagrange interpolation:

$$\begin{aligned}\psi_i^{n+1} &= (1-\alpha)\,\psi_{i-p}^n + \alpha\psi_{i-p-1},\\ &= \mathsf{S}^{-\mathrm{p}}\left[(1-\alpha)+\alpha\mathsf{S}^{-1}\right]\psi_i^n,\end{aligned}$$

where the shift operator S^{-p} indicates that we have to inversely shift p grid points from the nearest grid point to the departure point to the arrival point index at $t = n+1$. This then makes the numerical evolution operator for the linear interpolation semi-Lagrangian E_1 as

$$\mathsf{E}_1 = \mathsf{S}^{-\mathrm{p}}\left[(1-\alpha)+\alpha\mathsf{S}^{-1}\right]. \tag{10.90}$$

The associate Fourier symbol F_1 for the linear interpolation semi-Lagrangian scheme is derived by considering the effects that the evolution operator E_1 has on the discrete wave function $e^{ij\varphi}$, where j is a spatial index, as

$$\mathsf{F}_1 \equiv \mathsf{E}_1 e^{ij\varphi} = e^{-ip\varphi}\left[(1-\alpha)+\alpha e^{-i\varphi}\right], \tag{10.91}$$

where we have substituted the number of spatial grid points inversely moved from the index for the arrival point to the index of the nearest grid point upwind of the departure point which is equivalent to $-\mathrm{p}$. The -1 power in the second exponential term in (10.91) comes from the shift from the nearest point to the departure point upwind to the first grid point downwind of the departure point.

To start the stability analysis of the linear interpolation semi-Lagrangian scheme, we need to recall the amplification factor of the first-order upwind scheme, which is

$$1+\alpha+\alpha e^{-i\varphi},$$

which has the condition that $\overline{c}\frac{\Delta t}{\Delta x} \leq 1$ and is equivalent to $\alpha \leq 1$. We have normalized the distance between the grid points by Δx in α, which means that α cannot be greater than or equal to one as this would add an integer of 1 to P. This then means that we can satisfy the equivalent amplification factor constraint for the non-integer component of the Courant number. We now turn our attention to the integer component of the Courant number in (10.91) and recall that the stability of a numerical scheme is dependent on the modulus of the **real** component being less than or equal to one.

The term associated with the inverse shift of p grid points in (10.91), given by $e^{-ip\varphi}$, whose real number component is 1. Therefore, the modulus of the first exponential component in (10.91) is always equal to one (i.e., $|1| = 1$).

Therefore, the normalization by Δx and the fact that the extra term in the amplification factor in (10.91) has a modulus equal to one for all Δt and Δx implies that the linear semi-Lagrangian schemes is **unconditionally stable**. This is an important property of semi-Lagrangian methods.

10.4.2 Stability Analysis of the Quadratic Lagrange Interpolation

We now consider the stability of the quadratic interpolation semi-Lagrangian scheme which, as with the linear case, starts with defining the quadratic interpolation semi-Lagrangian difference equation in terms of the shift operator, which is

$$\begin{aligned}\psi_i^{n+1} &= \frac{\alpha\,(\alpha+1)}{2}\psi_{i-p-1}^n + \left(1-\alpha^2\right)\psi_{i-p}^n + \frac{\alpha\,(\alpha-1)}{2}\psi_{i-p+1},\\ &\equiv \mathsf{S}^{-p}\left[\frac{(\alpha^2+\alpha)}{2}\mathsf{S}^{-1} + \left(1-\alpha^2\right) + \frac{(\alpha^2-1)}{2}\mathsf{S}^{+1}\right]\psi_i^n,\\ \Rightarrow \mathsf{E}_2 &\equiv \mathsf{S}^{-p}\left[\frac{(\alpha^2+\alpha)}{2} + \left(1-\alpha^2\right) + \frac{(\alpha^2-\alpha)}{2}\right].\end{aligned} \tag{10.92}$$

Given the evolution operator for the quadratic interpolation-based semi-Lagrangian approach, E_2 we now move to define the Fourier symbol for this configuration. As with the linear interpolation case, we apply the quadratic evolution operator to the discrete wave function $e^{ij\varphi}$, which results in

$$\mathsf{F}_2 = \mathsf{E}_2 e^{ij\varphi} \equiv e^{-ip\varphi}\left[\frac{(\alpha^2+\alpha)}{2}e^{-i\varphi} + \left(1-\alpha^2\right) + \frac{(\alpha^2-\alpha)}{2}e^{i\varphi}\right]. \tag{10.93}$$

The next step in the stability analysis is to collect like terms in powers of α, which results in

$$\begin{aligned}\mathsf{F}_2 &= e^{-ip\varphi}\left(1-\alpha^2\left(1-\frac{1}{2}\left(e^{-i\varphi}+e^{i\varphi}\right)\right)+\frac{\alpha}{2}\left(e^{-i\varphi}-e^{i\varphi}\right)\right),\\ &\equiv e^{-ip\varphi}\left(1-\alpha^2\,(1-\cos\varphi) - i\alpha\sin\varphi\right).\end{aligned} \tag{10.94}$$

To complete the stability analysis for the quadratic interpolation semi-Lagrangian scheme, we need to consider the modulus and argument of (10.94) and see not only if they are bounded but also if there are any constraints on the step sizes.

Considering the modulus of F_2 first, we have

$$\begin{aligned}|\mathsf{F}_2| &= \left(1-\alpha^2\,(1-\cos\varphi)\right)^2 + \alpha^2\sin^2\varphi,\\ &= 1-2\alpha^2\,(1-\cos\varphi) + \alpha^4\,(1-\cos\varphi)^2 + \alpha^2\left(1-\cos^2\varphi\right),\\ &= 1-\alpha^2+2\alpha^2\cos\varphi - \alpha^2\cos^2\varphi + \alpha^4\,(1-\cos\theta)^2,\\ &= 1-\alpha^2\left((1-\cos\varphi)^2 - \alpha^2\,(1-\cos\varphi)\right),\\ &= 1-\alpha^2\left(1-\alpha^2\right)(1-\cos\varphi)^2.\end{aligned} \tag{10.95}$$

To ensure stability we require (10.95) to be less than or equal to 1, which implies that we have the following inequality:

$$-1 \le 1-\alpha^2\left(1-\alpha^2\right) \le 1 \Rightarrow -1 \le \alpha \le 1. \tag{10.96}$$

The condition for the rational component of the Courant number is always satisfied due to the restriction on the distance between the departure point and the nearest upwind grid point has been normalized by the spatial step size; therefore, the quadratic interpolation semi-Lagrangian scheme is unconditionally stable, the same as the linear interpolation.

In Figs. 10.3 and 10.4 we have plotted the modulus of the linear and quadratic Fourier symbol against the rational component of the Courant number for different wavelengths. These plots are recreations of those from [27] where these plots are explained as showing the scales that are damped by

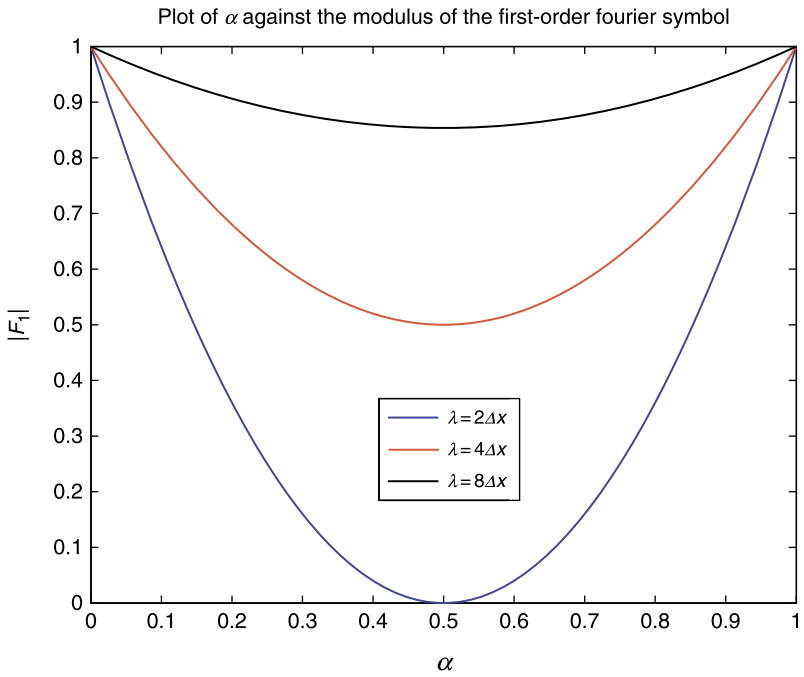


FIGURE 10.3

Plot of the amplification factor against the rational part of the Courant number for wave lengths $2\Delta x$, $4\Delta x$, and $8\Delta x$, respectively, for the linear Lagrange interpolation polynomial.

the different order interpolation schemes. In [27] they state that for the semi-Lagrangian scheme with linear interpolation then there is heavy damping for the shortest wavelengths, whereas for the shortest resolvable wave at $\lambda = 2\Delta x$ there is complete extinction of this wave when $\alpha = 0.5$. However, the damping effect becomes less serve as the wavelength increases. If we consider the associated plot for the semi-Lagrangian scheme with quadratic interpolation, Fig. 10.4, then we can see that while there is still extinction of the short waves here, it does not occur until $\alpha \pm \frac{1}{\sqrt{2}}$ and quickly dissipates for the larger wavelengths.

10.4.3 Stability Analysis of the Cubic Lagrange Interpolation

If we now consider the cubic interpolation semi-Lagrangian scheme, then upon multiplying out all of the brackets it is possible to write the polynomial in terms of powers of α as

$$\psi_i^{n+1} = \psi_{i-p}^n - \frac{1}{2}\alpha\left(\psi_{i-p+1}^n - \psi_{i-p-1}^n\right), + \frac{\alpha^2}{2}\left(\psi_{i-p+1}^n - 2\psi_{i-p}^n + \psi_{i-p-1}^n\right) + \frac{\alpha\left(1-\alpha^2\right)}{6}\left(\psi_{i-p+1}^n - 3\psi_{i-p}^n + 3\psi_{i-p-1}^n - \psi_{i-p-2}^n\right). \tag{10.97}$$

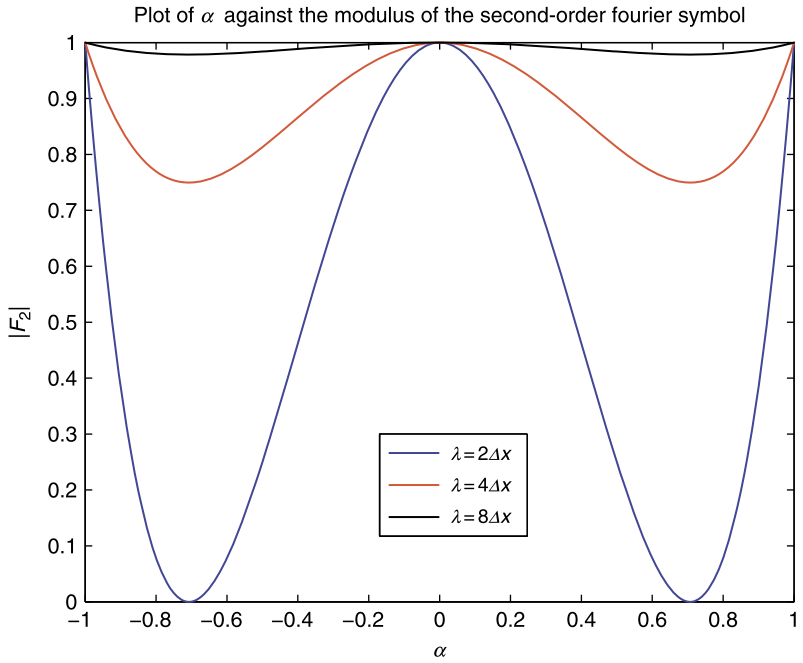


FIGURE 10.4

Plot of the amplification factor against the rational part of the Courant number for wave lengths $2\Delta x$, $4\Delta x$, and $8\Delta x$, respectively, for the quadratic Lagrange interpolation polynomial.

Next we express (10.97) in terms of the shift operator as

$$\psi_i^{n+1} = \mathsf{S}^{-p}\left[1 - \frac{1}{2}\alpha\left(\mathsf{S}^{+1} - \mathsf{S}^{-1}\right) + \frac{\alpha^2}{2}\left(\mathsf{S}^{+1} - 2 + \mathsf{S}^{-1}\right) - \frac{\alpha\left(1-\alpha^2\right)}{6}\left(\mathsf{S}^{+1} - 3 + 3\mathsf{S}^{-1} - \mathsf{S}^{-2}\right)\right]\psi_i^n. \tag{10.98}$$

Therefore, the evolution operator for the cubic Lagrange interpolation-based semi-Lagrangian scheme, E_3, is

$$\mathsf{E}_3 = \mathsf{S}^{-p}\left[1 - \frac{1}{2}\alpha\left(\mathsf{S}^{+1} - \mathsf{S}^{-1}\right) + \frac{\alpha^2}{2}\left(\mathsf{S}^{+1} - 2 + \mathsf{S}^{-1}\right) - \frac{\alpha\left(1-\alpha^2\right)}{6}\left(\mathsf{S}^{+1} - 3 + 3\mathsf{S}^{-1} - \mathsf{S}^{-2}\right)\right].$$

Now we apply the third-order numerical evolution operator to the Fourier node, $e^{ij\varphi}$, which results in

$$\mathsf{F}_3 \equiv \mathsf{E}_3 e^{ik\varphi} = e^{-ip\varphi}\left[1 - \frac{\alpha}{2}\left(e^{i\varphi} - e^{-i\varphi}\right) + \frac{\alpha^2}{2}\left(e^{i\varphi} - 2 + e^{-i\varphi}\right) + \frac{\alpha\left(1-\alpha^2\right)}{6}\left(e^{i\varphi} - 3 + 3e^{-i\varphi} - e^{-2i\varphi}\right)\right]. \tag{10.99}$$

To obtain the amplification factor we require $\left|\mathsf{F}_3\right|^2$. We start by replacing the exponentials in (10.99) with their relative sine and cosine equivalents, which yields

$$\begin{aligned}\mathsf{F}_3 &= e^{-ip\varphi}\left[1 - i\alpha\sin\varphi - \alpha^2(1-\cos\varphi) + \frac{\alpha\left(1-\alpha^2\right)}{6}\left(4\cos\varphi - 2i\sin\varphi + 2 - 2\cos^2\varphi - 2i\cos\varphi\sin\varphi\right)\right],\\ &= e^{-ip\varphi}\left[1 - i\alpha\sin\varphi - \alpha^2(1-\cos\varphi) - \frac{\alpha\left(1-\alpha^2\right)}{3}\left((1-\cos\varphi)^2 + i\sin\varphi(1-\cos\varphi)\right)\right].\end{aligned} \tag{10.100}$$

Collecting the real and imaginary parts results in

$$\mathrm{Re}\left(\mathsf{F}_3\right) = 1 - \alpha^2(1-\cos\varphi) - \frac{\alpha\left(1-\alpha^2\right)}{3}(1-\cos\varphi)^2, \tag{10.101}$$

$$\mathrm{Im}\left(\mathsf{F}_3\right) = -\alpha\sin\varphi - \frac{\alpha\left(1-\alpha^2\right)}{3}\sin\varphi(1-\cos\varphi). \tag{10.102}$$

The modulus of the Fourier symbol for the cubic interpolation is equivalent to the sum of the square of (10.101) and (10.102). Therefore, squaring the two parts above results in

$$\begin{aligned}\left|\mathsf{F}_3\right| = 1 &+ \alpha^4(1-\cos\varphi)^2 + \frac{\alpha^2\left(1-\alpha^2\right)^2}{9}(1-\cos\varphi)^4 - 2\alpha^2(1-\cos\varphi)\\ &- \frac{2\alpha\left(1-\alpha^2\right)}{3}(1-\cos\varphi)^2 + \frac{2\alpha^3\left(1-\alpha^2\right)}{3}(1-\cos\varphi)^3\\ &+ \alpha^2\sin^2\varphi + \frac{\alpha^2\left(1-\alpha^2\right)^2}{9}\sin^2\varphi(1-\cos\varphi)^2 + \frac{2\alpha^2\left(1-\alpha^2\right)}{3}\sin^2\varphi(1-\cos\varphi).\end{aligned} \tag{10.103}$$

The expression in (10.103)may appear quite daunting; however, to help simplify things we expand the powers of $(1-\cos\varphi)$, which are

$$\begin{aligned}C^2 &= (1-c) = 1 - 2c + c^2,\\ C^3 &= (1-c)^3 = 1 - 3c + 3c^2 - c^3,\\ C^4 &= (1-c)^4 = 1 - 4c + 6c^2 - 4c^3 + c^4,\end{aligned}$$

where $c = \cos\varphi$. The reason for displaying these expansions is because the expression for the modulus of the third-order Fourier symbol is presented in [279], but in terms of $C = (1-\cos\varphi)$. The factorized

expression for the amplification error in [279] is given (10.104), and below that we have expanded the brackets to ascertain which expressions we are looking for to verify that we have derived the correct amplification factor. Thus

$$|\mathsf{F}_3| = 1 - \alpha(2-\alpha)\left(1-\alpha^2\right)C^2\left(\frac{3+2C\alpha(1-\alpha)}{9}\right), \tag{10.104}$$

$$= 1 - \left(2\alpha - \alpha^2 - 2\alpha^3 + \alpha^4\right)C^2\left(\frac{3+2C\left(\alpha-\alpha^2\right)}{9}\right),$$

$$= 1 - \frac{1}{9}\left(6\alpha C^2 - 3\alpha^2C^2 - 6\alpha^3C^2 + 3\alpha^4C^2 + 4\alpha^2C^3 - 2\alpha^3C^3 - 4\alpha^4C^3 + 2\alpha^5C^3 - 4\alpha^3C^3 + 2\alpha^4C^3 + 4\alpha^5C^3 - 2\alpha^6C^3\right). \tag{10.105}$$

The next step is to collect all the factors of the different powers of α in (10.105), which results in

$$\begin{aligned}
\alpha^1 &: \quad -\tfrac{6}{9}C^2 \\
\alpha^2 &: \quad \tfrac{3}{9}C^2 - \tfrac{4}{9}C^3 \\
\alpha^3 &: \quad \tfrac{6}{9}C^2 + \tfrac{2}{9}C^3 + \tfrac{4}{9}C^3 \\
\alpha^4 &: \quad -\tfrac{3}{9}C^2 + \tfrac{4}{9}C^3 - \tfrac{2}{9}C^3 \\
\alpha^5 &: \quad -\tfrac{2}{9}C^3 - \tfrac{4}{9}C^3 \\
\alpha^6 &: \quad \tfrac{2}{9}C^3.
\end{aligned} \tag{10.106}$$

As we can see above, we do not have a C^4 term in the final expression from [279], yet we do have in (10.103). This indicates that we have to find a way to eliminate the C^4 term. To do this we have to *complete the cube* on two different terms in (10.103). The first term that we consider is the quartic term and find a way to express the quartic as a sum of cube expressions. This is achieved by

$(1-c)^4$	1	$-4c$	$6c^2$	$-4c^3$	c^4
$(1-c)^3$	1	$-3c$	$3c^2$	$-c^3$	
	0	$-c$	$3c^2$	$-3c^3$	$-c^4$

where $c = \cos\varphi$ and the last line above is equivalent to $-c(1-c)^3$. Therefore, it is possible to write the quartic term as $(1-c)^4 = C^3 - cC^3$. The next term that we have to consider is $\left(1-c^2\right)(1-c)^2$.

$\left(1-c^2\right)(1-c)^2$	1	$-2c$		$2c^3$	$-c^4$
$(1-c)^3$	1	$-3c$	$3c^2$	$-c^3$	
	0	c	$-3c^2$	$3c^3$	$-c^4$

The last line in the array above is cC^3. Therefore, the cC^3 terms cancel and we have matched the α^6 term of (10.106).

We now consider the lower powers of α from (10.106). From (10.103) we already have the correct coefficient for α; however, for α^2 we have a term that is only a factor of C yet all of the coefficients in (10.106) are in terms of C^2 or C^3 and none of them are in terms of $\sin^2\varphi$. Therefore, collecting all the

factors of α^2 and using the trigonometric identity $\sin\varphi \equiv 1 - \cos^2\varphi$, and putting all of the coefficients under a common denominator, we have

$$\begin{aligned}\alpha^2 &: \sin^2\varphi + \frac{2}{3}\alpha^2\sin^2\varphi\,(1-\cos\varphi) - 2\,(1-\cos\varphi)\,,\\ &3 - 3c^2 + 2 - 2c^2 - 2c + 2c^3 - 6 + 6c,\\ &-1 + 4c - 5c^2 + 2c^3. \end{aligned} \tag{10.107}$$

The expression in (10.107) appear to be in the form of C^3, and if we complete the cube on (10.107) with $-2C^3$ we obtain

Eq. (10.107)	-1	$4c$	$-5c^3$	$2c^3$
$-2C^3$	-2	$6c$	$-6c^2$	$2c^3$
	1	$-2c$	c^2	

Therefore, the factors of the α^2 terms combine to $-\frac{6}{9}C^3 + \frac{3}{9}C^2$, which gives us the C^2 factor for α^2 in (10.106). To obtain the factor of C^3 we should note that what was the C^4 is now in terms of $2C^3$ and upon expanding the $\frac{2\alpha^2(1-\alpha^2)^2}{9}$ term we have a $\frac{2}{9}\alpha^2C^3$, which completes the factors of α^2 in (10.106).

Moving on to the α^3, we first need to notice that there are two factors of the C^3 term here, which we added together give a factor of $\frac{6}{9}C^3$. We can then see that both of the factors of α^3 are present in (10.103); therefore, we have matched the α^3 factors. However, for the α^4 factors we shall have to do some rearranging and completing the cube again. Combining all of the current factors of α^4, substituting for $\sin^2\varphi$ again, and putting all the terms under a common denominator results in

$$\begin{aligned}\alpha^4 &: 3\,(1-\cos\varphi)^2 - 2\left(1-\cos^2\varphi\right)(1-\cos\varphi)\,,\\ &3 - 6c + 3c^2 - 2 + 2c + 2c^2 - 2c^3,\\ &1 + 4c + 5c^2 - 2c^3,\\ &\Rightarrow -2C^3 - C^2. \end{aligned}$$

Next, putting the terms above under the other common denominator of 9 gives us the required factor of C^2 for α^4. Combining with the old C^4 term which also contains an α^4 of 4 results in the factor of C^3 when the two terms are added together in (10.106).

This now just leaves the α^5 factors to consider. There are no C^2s in the factors of α^5, so we are then left with two factors of C^3, which when added together we require $-\frac{6}{9}C^3$, upon looking at (10.103), we can see this is present. Thus we have verified the expression for the amplification factor of the cubic interpolation semi-Lagrangian scheme from [279]. Given the expression for the amplification factor in the factorized form we can see that the restrictions of the spatial and temporal step sizes is that $0 \leq \alpha < 1$, which comes about because if $\alpha < 0$ then the $-\alpha$ terms make the second term in (10.104) positive which then makes the amplification greater than one. If we consider the case where $\alpha \geq 1$ then we see that the amplification factor is either equal to one, which is not a desirable property, or is greater than one due to the one of the terms involving α will be negative and as such changes the sign of the second term in (10.104).

Given that we now know that the fractional part of the Courant number must satisfy $0 \leq \alpha < 1$, then as we stated for the linear case, this condition is always satisfied because the distance between grid points has been normalized to 1, therefore the cubic interpolation semi-Lagrange scheme is unconditionally stable.

This may appear to be a lot of work to obtain the amplification factor but we need this expression to ascertain if we will have the damping effects that we have seen for the linear and quadratic interpolation semi-Lagrangian schemes. In Fig. 10.5 we have plotted the amplification factor for the cubic interpolation.

As we can see from Fig. 10.5, the short wavelengths are still being damped to the point that if the departure point is halfway between two points, then we lose all information of that feature. However, when considering the longer wavelengths, we see that the cubic interpolation scheme does not damp as heavily as the linear interpolations does. This improvement is quite significant, and is one reason why the cubic interpolation-based scheme is used; while it does take more coding than the two lower-order interpolations and does take longer to run, it is equivalent to a third order in space discretization.

Exercise 10.9. *For the quartic Lagrange interpolation that uses the points* $\{i-p-2, i-p-1, i-p, i-p+1, i-p+2\}$*, derive the fourth-order evolution operator* E_4 *and then the fourth-order Fourier symbol,* F_4*. Show that the real and imaginary components of the Fourier symbol are*

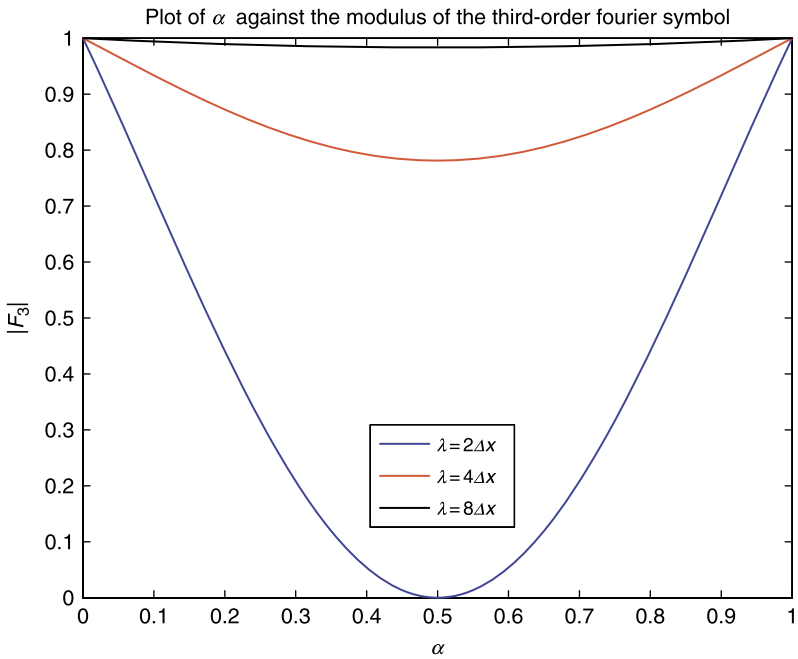


FIGURE 10.5

Plot of the amplification factor against the rational part of the Courant number for wave lengths $2\Delta x$, $4\Delta x$, and $8\Delta x$, respectively, for the cubic Lagrange interpolation polynomial.

$$Re\,(F_4) = 1 - C\alpha^2 - \frac{C^2\alpha^2\left(1-\alpha^2\right)}{6},$$
$$Im\,(F_4) = \alpha \sin\varphi \left(1 + \frac{C\left(1-\alpha^2\right)}{3}\right).$$

By taking the modulus of F_4, show that the amplification factor can be factorized to

$$|F_4| = 1 - \alpha^2\left(1-\alpha^2\right)\left(4-\alpha^2\right)C^3\frac{\left(C\left(1-\alpha^2\right)\right)}{36}. \tag{10.108}$$

Given the amplification factor in (10.108) determine the values for α that the quartic Lagrange interpolation scheme is stable and deduce that the type of stability for the scheme.

The expressions above as we have stated are from [279] where they introduce the extra constraint on α for the quadratic and quartic schemes so that the rational component of the Courant number, or rather the normalized (by Δx) distance from the departure point cannot be more than ± 0.5. We can clearly see the reason for this from Fig. 10.4, where if we have $-0.5 \leq \alpha \leq 0.5$, then the scheme will not completely damp the smaller wavelengths and is in fact less damping on the longer wavelengths than the cubic interpolation [279].

10.4.4 Stability Analysis of the Cubic Hermite Semi-Lagrangian Interpolation Scheme

As we have observed with the different order of Lagrange polynomial-based interpolation schemes for the semi-Lagrangian advection case, there are two different ways to factorize the interpolation polynomial. The first is to collect factors of different powers of the Courant number, while the second approach is to collect the factors of the grid points. We are going to consider the latter case for a specific second-order version of the derivatives in a Hermite interpolation formulation.

The second-order approximation to the derivative scheme that we present here comes from [239], where the derivatives at the end points are approximated by $\frac{\psi_{i-p}-\psi_{i-p-2}}{2}$ for ψ'_{i-p-1} and by $\frac{\psi_{i-p+1}-\psi_{i-p-1}}{2}$ for ψ'_{i-p}. Therefore, the Hermite interpolation polynomial is going to be using the same points as the cubic Lagrange interpolation polynomial, but here with different factors. We should note that this is equivalent to the arithmetic mean approximation to the derivatives mentioned in Section 10.3.3.

In [239] it is stated that the 1D Hermite interpolation scheme has an interval $x \in (x_{i-1}, x_i)$ of length Δx, but where we have normalized the distance between the grid points in terms of the Courant number, this then gives the general form of the cubic Hermite interpolation formula as

$$\begin{aligned}\psi\,(\tilde{x}) = {} & (1-\alpha)^3\left(2\psi_{i-p-1} - 2\psi_{i-p} + \psi'_{i-p-1} + \psi'_{i-p}\right) \\ & + (1-\alpha)^2\left(-3\psi_{i-p-1} + 3\psi_{i-p} - 2\psi'_{i-p-1} - \psi'_{i-p}\right) + (1-\alpha)\,\psi'_{i-1} + \psi_{i-1}.\end{aligned} \tag{10.109}$$

Substituting the second-order approximations to the derivative terms at the end points into (10.109) and collecting all the factors of the four grid points results in

$$\psi_i^{n+1} = g_{-2}\psi_{i-p-2}^n + g_{-1}\psi_{i-p-1}^n + g_0\psi_{i-p}^n + g_{+1}\psi_{i-p+1}^n,$$

where the coefficients above are defined as

$$
\begin{aligned}
g_2 &\equiv -\frac{1}{2}(1-\alpha)^3 + (1-\alpha)^2 - \frac{1}{2}(1-\alpha), \\
g_{-1} &\equiv \frac{3}{2}(1-\alpha)^3 - \frac{5}{2}(1-\alpha)^2 + 1, \\
g_0 &\equiv -\frac{3}{2}(1-\alpha)^3 + 2(1-\alpha)^2 + \frac{1}{2}(1-\alpha), \\
g_{+1} &\equiv \frac{1}{2}(1-\alpha)^3 - \frac{1}{2}(1-\alpha)^2.
\end{aligned}
$$

To perform the Van Neumann stability analysis for (10.109), we first identify the evolution operator;

$$
\mathsf{E}_{cH} = \mathsf{S}_{-p}\left[g_{-2}\mathsf{S}^{-2} + g_{-1}\mathsf{S}^{-1} + g_0 + g_1\mathsf{S}^{+1}\right],
$$

which would then make the Fourier symbol for the second-order derivative base cubic Hermite interpolation:

$$
\mathsf{F}_{cH} = \mathsf{E}_{cH} a_n e^{ik\varphi} \equiv e^{-pi\varphi}\left[g_{-2}e^{-2i\varphi} + g_{-1}e^{-i\varphi} + g_0 + g_{+1}e^{i\varphi}\right]. \tag{10.110}
$$

If we simply wish to examine the amplification factor for the cubic Hermite interpolation, it is possible to write a short piece of code to run through different values of α between 0 and 1 in the modulus of (10.110) to see where the amplification factor goes above one. In Fig. 10.6A we have plotted the amplification factor against the Courant number for different wavelengths to see which scales are damped with this scheme, while in Fig. 10.6B we have plotted the amplification factor against wavelengths for

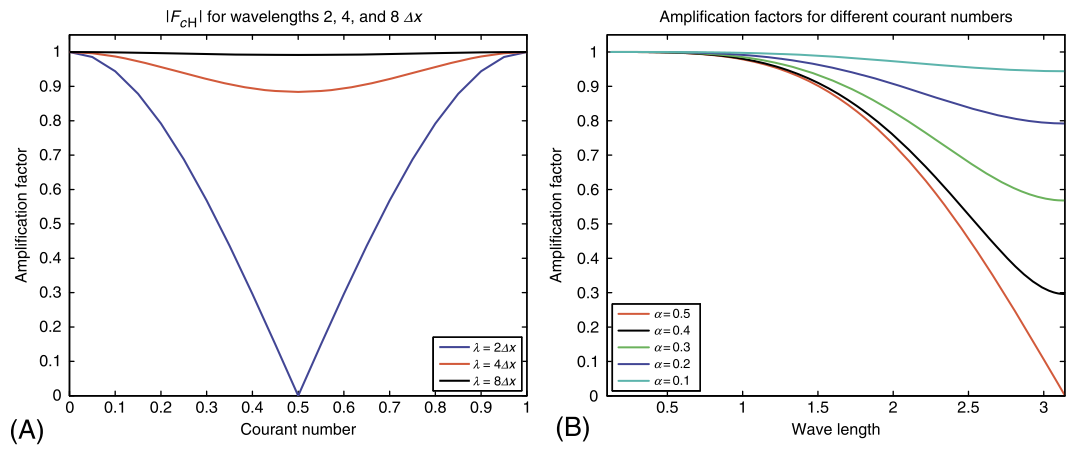


FIGURE 10.6

(A) Plot of the amplification factor against Courant number for the cubic Hermite interpolation scheme for different wavelengths; (B) plot of amplification factors by Courant number against the wavelength for the cubic Hermite interpolation scheme.

different Courant numbers to see which wavelengths are damped due to the distance of the departure point to the nearest grid point.

We can see from Fig. 10.6A that the cubic Hermite approximations are stable for Courant numbers between 0 and 1, which means that it is again unconditionally stable; we can see this for the three wavelength amplification curves that we have plotted. As with the cubic Lagrange interpolation, we see that for wavelengths $2\Delta x$; if the Courant number is $\alpha = 0.5$, then this feature would again be completely damped. However, the structure of the curve associated with $k = 2\Delta x$ for the other values of the Courant number is not as severe.

If we now analyze the response to the amplification for the $k = 4\Delta x$ wavelength, then we see that there is a drastic improvement over the performance of the Hermite polynomial for the $k = 2\Delta x$ case. While there is still some damping when the departure point is in the center of the interval, the wave feature will not be too severely damped. When we look at the amplification factor for the $k = 8\Delta x$ case, we see that there is almost no damping at this wavelength.

As mentioned earlier, in Fig. 10.6B we have plotted the amplification factors by Courant number against the wavelength, which is another useful way to identify where the interpolation schemes are not introducing too much of an error into the numerical scheme. We see that for $\alpha = 0.5$, as the wavelength reaches π, the scheme damps this feature to zero. As the Courant numbers moves away from $\alpha = 0.5$, the damping effects reduce quite quickly, suggesting that if the departure point is away from the center part of the interval, then the interpolation is drastically improved.

If, however, we wished to determine the stability analytically for this version of the cubic Hermite interpolation, then we would start by expanding the coefficients in (10.110) into powers of α and collecting all the factors of each of the four powers of α results in

$$\begin{aligned}
\alpha^3: &\quad \frac{e^{-2i\varphi}}{2} - \frac{3e^{-i\varphi}}{2} + \frac{3}{2} - \frac{e^{i\varphi}}{2}, && (1-C)^2 + i\sin\varphi\,(1-C)\,,\\
\alpha^2: &\quad -\frac{e^{-2i\varphi}}{2} + 2e^{-i\varphi} - \frac{5}{2} + e^{i\varphi}, && -(1-C)^2 - i\sin\varphi\,(1-C) - (1-C)\,,\\
\alpha^1: &\quad \frac{e^{-i\varphi}}{2} - \frac{e^{i\varphi}}{2}, && -i\sin\varphi,\\
\alpha^0: &\quad 1, && 1.
\end{aligned}$$

Combining all of the information above into the real and imaginary part of the Fourier symbol results in

$$\mathrm{Re}\left(\mathsf{F}_{cH}\right) = 1 - \left((1-C) + (1-C)^2\right)\alpha^2 + (1-C)^2\alpha^3, \tag{10.111a}$$

$$\mathrm{Im}\left(\mathsf{F}_{cH}\right) = -i\sin\varphi\alpha - i\sin\varphi\,(1-C)\,\alpha^2 + i\sin\varphi\,(1-C)\,\alpha^3. \tag{10.111b}$$

Therefore, the amplification factor of the cubic Hermite interpolation is the modulus of the sum of (10.111a) and (10.111b).

As mentioned before, for the derivation above we now do a comparison of the amplification factors of the cubic Lagrange interpolation with the cubic Hermite interpolation with the arithmetic mean/central difference approximation for the gradient terms. We have presented the amplification factors for the two schemes in the same way that we have in Fig. 10.6 in Fig. 10.7.

As with Fig. 10.6A we have plotted the amplification factor by Courant number for different wave numbers for comparison between the Lagrange interpolation and the Hermite polynomial with the

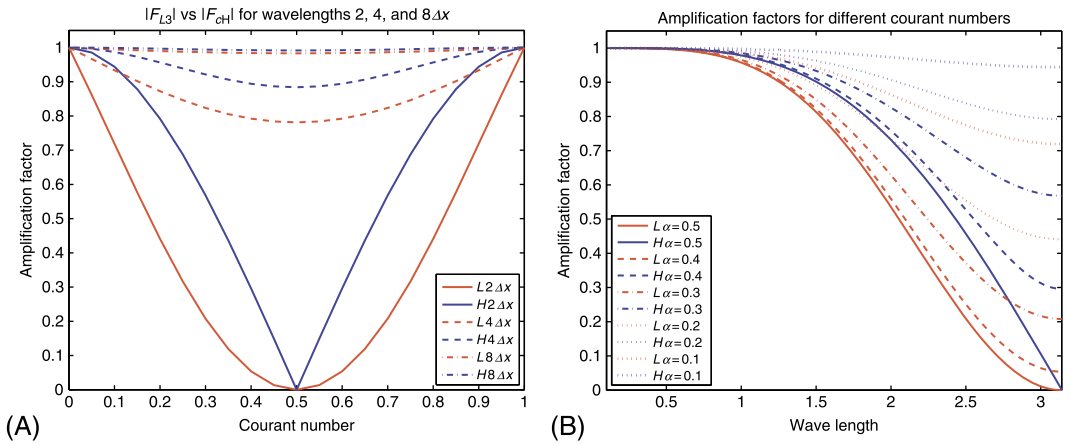


FIGURE 10.7

(A) Plot of the amplification factor against Courant number for the cubic Hermite and Lagrange interpolation schemes for different wavelengths; (B) plot of amplification factors by Courant number against the wavelength for the cubic Hermite and Lagrange interpolation schemes.

central difference approximation in Fig. 10.7A. The first feature to note here is that while both the Lagrange and the Hermite polynomial both damp out the wave number $2\Delta x$ features, the Hermite interpolation that we are considering in this stability analysis does so at quite a slower rate than the cubic Lagrange interpolation polynomial. For the Lagrange interpolation we see that the damping effects start immediately for the departure points that are not at a grid point, but for the Hermite interpolation we see that the damping is less severe for all values of the Courant number, except when the departure point is at the center of the interval.

Moving on to the results for wave number $k = 4\Delta x$, we see that while both schemes have drastically reduced the damping feature seen for the $2\Delta x$ case, there is still some damping; however, this is clearly smaller for the Hermite interpolation than for the Lagrange interpolation. For the wavenumber $8\Delta x$ case we again see a reduction in the damping effect. However, there is a slight improvement at this number with the Hermite interpolation over using the Lagrange interpolation.

In Fig. 10.7B we have plotted the amplification factor by wave length for the $\alpha = 0.1, 0.2, 0.3, 0.4,$ and 0.5 for both the cubic Lagrange polynomial and our choice for the cubic Hermite polynomials. We can see that the Hermite polynomial damping effect is always smaller than that with the Lagrange interpolation. Another striking feature of Fig. 10.6B is the difference between the amplification factors of the two schemes as the Courant number decreases toward zero. The amplification factor for the Hermite interpolation when $\alpha = 0.4$ is 2.5 units larger than that of the equivalent Lagrange interpolation. The difference in the amplification factors grows even more when $\alpha = 0.3$. However, the difference between the two schemes does start to decrease for $\alpha = 0.2$; where we see that the version of the Hermite interpolation we consider in this section has less of a damping effect than the Lagrangian interpolation.

10.4.5 Stability Analysis of the Cubic Spline Semi-Lagrangian Interpolation Scheme

The stability for the cubic spline interpolation polynomial for the semi-Lagrangian advection problem is stated in [371]. The derivation of the Neumann stability for the cubic spline starts from writing the associated polynomial in terms of the shift operator as

$$\frac{1}{6}\psi_{i-1}^{n+1}+\frac{2}{3}\psi_{i}^{n+1}+\frac{1}{6}\psi_{i+1}^{n+1}=\mathsf{S}^{-p}\left[\mathsf{S}^{-2}c_{-2}+\mathsf{S}^{-1}c_{-1}+c_0+\mathsf{S}^{+1}c_1\right],$$

where the right-hand side of the equation above is at time n and would be our evolution operator for the cubic spline. Introducing the Fourier node $\psi_i^n = a_n e^{ik\varphi}$, where this time we have included the time component into the Fourier node due to the implicit nature of the cubic spline, which results in

$$\mathsf{F}_{cs}^{n}=\mathsf{E}_{cs}^{n}a_n e^{ij\phi}\equiv e^{-ip\varphi}\left(e^{-2i\varphi}c_{-2}+e^{-i\varphi}c_{-1}+c_0+e^{i\varphi}c_1\right), \tag{10.112a}$$

$$\begin{aligned}\mathsf{F}_{cs}^{n+1}&=a_{n+1}\left(\frac{1}{6}e^{-i\varphi}+\frac{2}{3}+\frac{1}{6}e^{i\varphi}\right),\\&\equiv a_{n+1}\left(\frac{2}{3}+\frac{1}{3}\cos\varphi\right).\end{aligned} \tag{10.112b}$$

Equating the two expressions in (10.112a) and (10.112b) and rearranging to have the time components on the left-hand side results in

$$\frac{a_{n+1}}{a_n}=\frac{c_{-2}e^{-2i\varphi}+c_{-1}e^{-i\varphi}+c_0+c_1e^{i\varphi}}{\frac{2}{3}+\frac{1}{3}\cos\varphi}. \tag{10.113}$$

The expression in (10.113) is referred to as the *symbol of the scheme* in [371] which is equivalent to our Fourier symbol terminology that we used for the other interpolation-based semi-Lagrangian advection schemes that we have presented.

To determine the amplification factor for the cubic spline-based semi-Lagrangian advection we need to take the modulus of (10.113). If we expand the c_i coefficients in (10.113), and collect the factors for the powers of α, then we have

$$\alpha^0\colon \frac{1}{6}\left(4+e^{i\varphi}+e^{i\varphi}\right)\equiv\frac{1}{3}\left(2+\cos\varphi\right), \tag{10.114a}$$

$$\alpha^1\colon \frac{1}{6}\left(-3e^{-i\varphi}+3e^{i\varphi}\right)\equiv -i\sin\varphi, \tag{10.114b}$$

$$\alpha^2\colon \frac{1}{6}\left(-6+3e^{i\varphi}+3e^{-i\varphi}\right)\equiv -\left(1-\cos\varphi\right), \tag{10.114c}$$

$$\alpha^3\colon \frac{1}{6}\left(3-e^{i\varphi}-3e^{-i\varphi}+e^{-2i\varphi}\right)\equiv\frac{1}{3}\left((1-\cos\varphi)^2+i\left(1-\cos\varphi\right)\sin\varphi\right). \tag{10.114d}$$

Combining the real and imaginary components from (10.114a)–(10.114d) results in

$$\mathrm{Re}\left(\mathsf{F}_{cs}\right)=\frac{1}{3}\left(2+\cos\varphi\right)-\left(1-\cos\varphi\right)\alpha^2+2\left(1-\cos\varphi\right)^2\alpha^3, \tag{10.115a}$$

$$\mathrm{Im}\left(\mathsf{F}_{cs}\right)=-\sin\varphi+2\left(1-\cos\varphi\right)\sin\varphi, \tag{10.115b}$$

which we can see is slightly different to the real and imaginary components of the Fourier symbol for the cubic Lagrange interpolation polynomial. To obtain the amplification factor we need to take the modulus of (10.115a) with (10.115b) divided by (10.112b); however, we should note that the first term in (10.115a) is the same as that in (10.112b). The modulus of a quotient of complex numbers is equal to the quotient of the modulus, therefore for the amplification factor for the cubic spline is given by

$$|\mathsf{F}_{cs}| \equiv \left| \frac{\mathrm{Re}\left(\mathsf{F}_{cs}\right) + i\mathrm{Im}\left(\mathsf{F}_{cs}\right)}{\frac{2}{3} + \frac{1}{3}\cos\varphi} \right| \equiv \frac{\left|\mathrm{Re}\left(\mathsf{F}_{cs}\right) + i\mathrm{Im}\left(\mathsf{F}_{cs}\right)\right|}{\left|\frac{2}{3} + \frac{1}{3}\cos\varphi\right|}. \tag{10.116}$$

It is left as an exercise to finish the derivation of the amplification factor for a cubic spline-based semi-Lagrangian advection scheme, but there is a detailed analysis of the properties of the cubic splines in [371], and we shall provide a brief summary their findings for this scheme.

We have already seen that the cubic Lagrange interpolation scheme completely damps waves of wavelength $2\Delta x$ when the Courant number, α, is equal to 0.5. In [371] the authors reiterate this finding from [279] but compare this to the properties of the cubic spline approach. We have recreated Fig. 5A from [371] in Fig. 10.8, but we have also recreated Fig. 4A from that paper to illustrate the differences in the amplification factors for the cubic spline and cubic Lagrange. As mentioned in [371], the cubic spline's amplification factor associated with $\alpha = 0.5$ does not have as much of a damping effect on the shorter wavelengths compared to the cubic-Lagrange approach. It is clear from Fig. 10.8B that the cubic spline approach has a very sharp decline in its amplification factor to the complete damping of the wave, while the cubic Lagrange covers a larger range of wavelengths that are damped in some form.

To round out the stability section of the semi-Lagrangian schemes, we have sets of figures associated with amplification factors of the schemes that we have introduced in this chapter. We have amplification factors for the linear, quadratic, and cubic Lagrange interpolations, along with the cubic Hermite polynomial and the cubic spline for wave numbers $2\Delta x$, $4\Delta x$, and $8\Delta x$ in Fig. 10.9A, B, and C, respectively. Taking the smallest wave number case first, it is clear that the Lagrange interpolation formulas

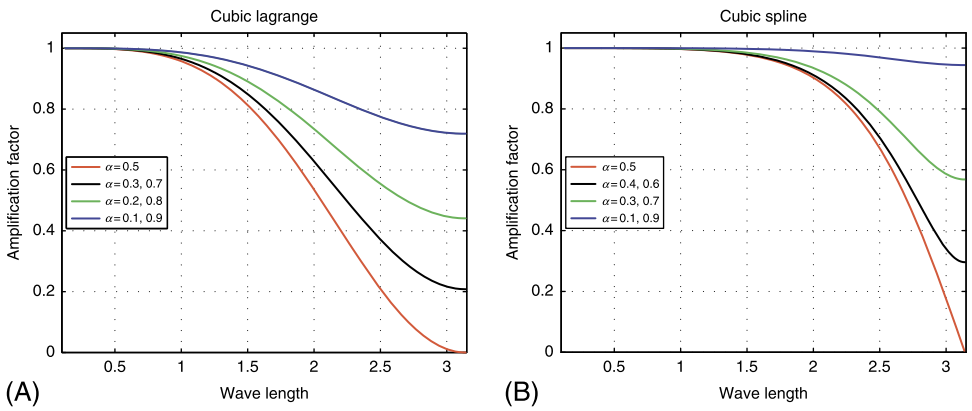


FIGURE 10.8

Recreation of the stability plot from Rushøjgaard et al. [371] where (A) is for the cubic Lagrange and (B) is for the cubic spline.

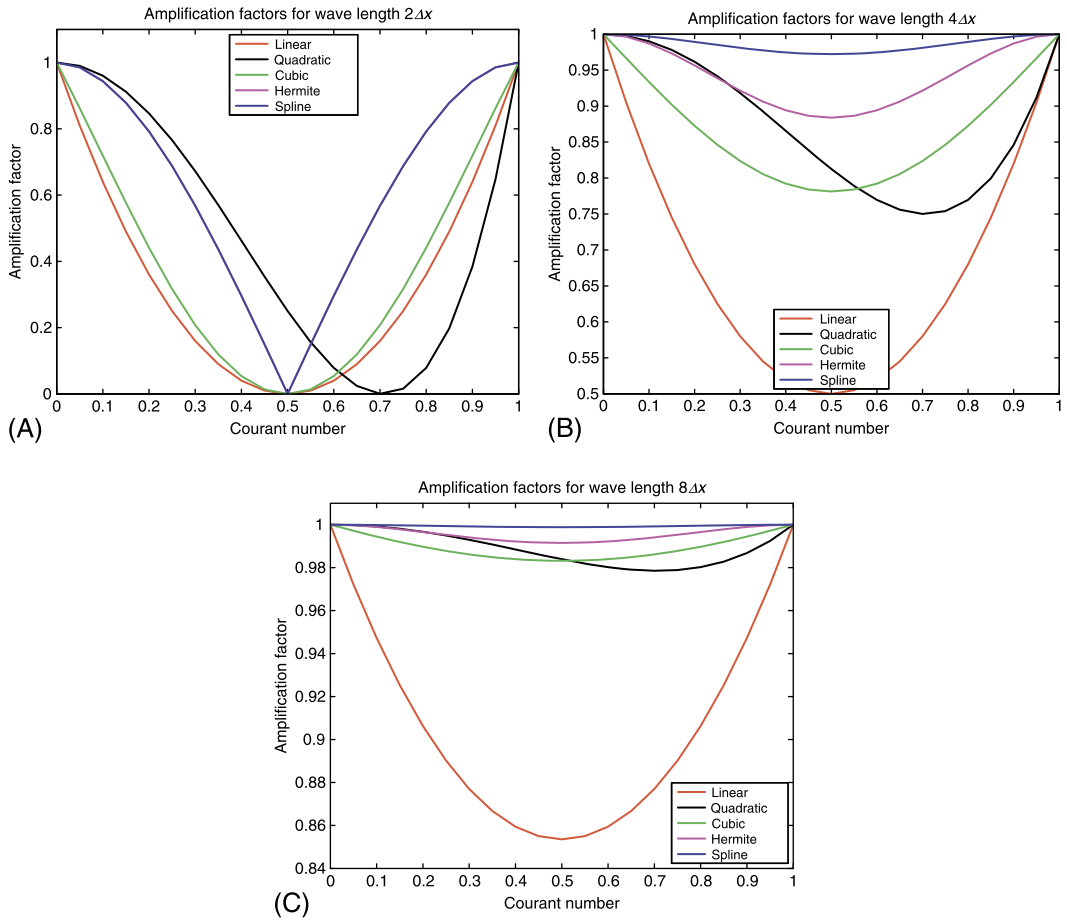


FIGURE 10.9

Amplification factors for the linear, quadratic, and cubic Lagrange polynomials along with cubic Hermite and the Cubic spline for (A) $k = 2\Delta x$, (B) $k = 4\Delta x$, and (C) $k = 8\Delta x$.

do not perform as well as the cubic Hermite and spline polynomials, and while it might appear that one of the schemes is missing, this is not true. For the $2\Delta x$ case it appears that the cubic Hermite and spline have the same amplification factors.

When we consider wavenumber $k = 4\Delta x$, we see that all of the schemes have reduced the damping effects from wavenumber $2\Delta x$, but the linear scheme still has a strong damping effect. We should note here that the scale on the y axis now starts at 0.5 and not 0; the reason for this scale is to highlight the difference between the other schemes, where we can clearly see that the Hermite and the spline approaches have drastically smaller damping effects that any of the Lagrange interpolations that we have considered. However, we should note that the damping effect associated with the cubic spline approach is still a lot smaller than the Hermite polynomial.

Finally, Fig. 10.9C is for wave number $k = 8\Delta x$ where we have again reduced the scale on the y-axis so that we can highlight the differences between the difference schemes. Therefore, we can clearly see that nearly all of the schemes, except for the linear Lagrange polynomial case (which is still not that bad), have nearly eliminated their damping effects. However, as with the other cases, we see that the cubic spline approach still appears to be best out of the methods that we have introduced.

In Fig. 10.10 we have plotted the amplification factors by wavelength for the three cubic interpolation polynomials for $\alpha = 0.5$. As we saw earlier, all three schemes damp out the features of wavelength π, but what is interesting is how each scheme approaches that wavelength. We can see that the most common damping scheme is the cubic Lagrange, followed by the cubic Hermite and then the cubic spline approach. This indicates that as we introduce higher-order constraints on the interpolation polynomial, the damping effects are reduced, but not eliminated.

However, just because we know that the semi-Lagrangian schemes that we have presented are unconditionally stable and can take larger time steps than the Eulerian schemes, we still have to address how accurate the semi-Lagrangian schemes are. Therefore, we now move on to consistency analysis of semi-Lagrangian schemes.

10.5 Consistency Analysis of Semi-Lagrangian Schemes

As we saw with the finite difference approximation to different initial value and boundary value ordinary and partial differential equations problems, we need to assess how accurate the semi-Lagrangian

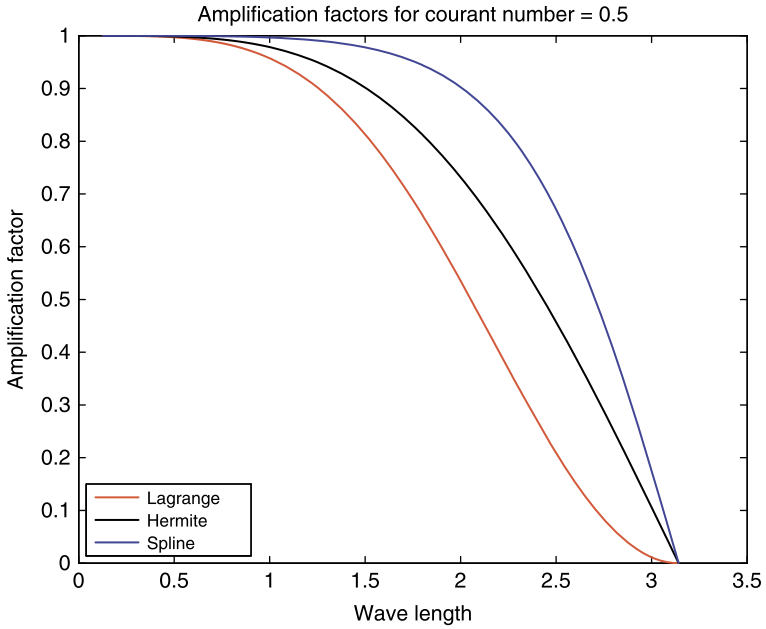


FIGURE 10.10

Plot of the amplification factors for the three cubic schemes against wave number.

approximation is. We shall only consider the consistency analysis for the quadratic Lagrange interpolation polynomial here, but the techniques are extendable to higher-order Lagrange interpolation polynomials and the other types of interpolation that we have mentioned in this chapter.

We recall the equation for the quadratic Lagrange interpolation semi-Lagrangian scheme

$$\psi_i^{n+1} = \frac{1}{2}\alpha\,(1+\alpha)\,\psi_{i-p-1}^n + (1-\alpha)\,(1+\alpha)\,\psi_{i-p}^n - \frac{1}{2}\alpha\,(1-\alpha)\,\psi_{i-p+1}^n, \tag{10.117}$$

where we have restricted α such that $-\frac{1}{2} \leq \alpha \leq \frac{1}{2}$. We have already shown that a quadratic scheme in this form is unconditionally stable for all choices of Δt and Δx, but we have not addressed the question of how accurate the scheme is.

To answer this question, we need to first show that the scheme is consistent with the Lagrangian/Eulerian advection differential equation and then determine the order of the remainder term. As with the Eulerian-based schemes we consider a Taylor series expansion of the tracer field, ψ, about the arrival point but at time n not $n+1$. Therefore, we have to expand all three terms in the quadratic interpolation formula with respect to x_i, which yields

$$\psi_{i-p-1}^n \approx \left(\psi_i - (p+1)\,\Delta x\left[\frac{\partial \psi}{\partial x}\right]_i^n + \frac{(p+1)^2}{2}(\Delta x)^2\left[\frac{\partial^2 \psi}{\partial x^2}\right]_i^n + \frac{(-(p+1))^3}{3!}(\Delta x)^3\left[\frac{\partial^3 \psi}{\partial x^3}\right]_i^n\right), \tag{10.118a}$$

$$\psi_{i-p}^n \approx \left(\psi_i - (p)\,\Delta x\left[\frac{\partial \psi}{\partial x}\right]_i^n + \frac{(p)^2}{2}(\Delta x)^2\left[\frac{\partial^2 \psi}{\partial x^2}\right]_i^n + \frac{(-(p))^3}{3!}(\Delta x)^3\left[\frac{\partial^3 \psi}{\partial x^3}\right]_i^n\right), \tag{10.118b}$$

$$\psi_{i-p+1}^n \approx \left(\psi_i + (-p+1)\,\Delta x\left[\frac{\partial \psi}{\partial x}\right]_i^n + \frac{(-p+1)^2}{2}(\Delta x)^2\left[\frac{\partial^2 \psi}{\partial x^2}\right]_i^n + \frac{((-p+1))^3}{3!}(\Delta x)^3\left[\frac{\partial^3 \psi}{\partial x^3}\right]_i^n\right). \tag{10.118c}$$

Combining the Taylor series expansions in (10.118a)–(10.118c) with their associated factors in (10.117) results in a Taylor series expansion for the tracer at the departure point as

$$\psi\,(x, n\Delta t) = \psi_i^n - (p-\alpha)\,\Delta x\left[\frac{\partial \psi}{\partial x}\right]_i^n + (p+\alpha)^2\,\frac{(\Delta x)^2}{2}\left[\frac{\partial^2 \psi}{\partial x^2}\right]_i^n + O\left(\Delta x^3\right), \tag{10.119}$$

where the leading third-order term in x is

$$O\left((\Delta x)^3\right) = \frac{(\Delta x)^3}{6}\left[\frac{\partial^3 \psi}{\partial x^3}\right]_i^n W_2^3,$$

where W_2^3 is defined as [279]

$$W_2^3\,(\alpha, p) \equiv \frac{1}{2}\alpha\,(1-\alpha)\,(-p-1)^3 + (1-\alpha)\,(1+\alpha)\,(-p)^3 - \frac{1}{2}\alpha\,(1-\alpha)\,(-p+1)^3\,. \tag{10.120}$$

The next stage in the derivation of the consistency check is to note that with $x = \tilde{x} = i\,\Delta x - \overline{u}\frac{\Delta t}{\Delta x}$; this then implies that $p + \alpha = \overline{u}\frac{\Delta t}{\Delta x}$. We now consider the forward upwind approximation to the semi-

Lagrangian form of advection as

$$\frac{\psi\left(i\Delta x,(n+1)\Delta t\right)-\psi\left(\tilde{x},n\Delta t\right)}{\Delta t}=0, \tag{10.121}$$

then substituting (10.119) evaluated at $p+\alpha=\overline{u}\frac{\Delta t}{\Delta x}$ yields

$$\psi\left(\tilde{x},n\Delta t\right)=\psi_i^n-\overline{u}\Delta t\left[\frac{\partial\psi}{\partial x}\right]_i^n+\overline{u}^2\left(\Delta t\right)^2\left[\frac{\partial^2\psi}{\partial x^2}\right]_i^n+O\left(\Delta t^3\right). \tag{10.122}$$

Next, expanding the term $\psi\left(i\Delta x,(n+1)\Delta t\right)$ about $i\Delta x,n\Delta t$ as a Taylor series results in

$$\psi\left(x_i,t^{n+1}\right)\approx\psi_i^n+\Delta t\left[\frac{\partial\psi}{\partial t}\right]_i^n+\frac{\Delta t^2}{2}\left[\frac{\partial^2\psi}{\partial t^2}\right]_i^n+O\left(\Delta t^3\right). \tag{10.123}$$

Substituting (10.122) and (10.123) into (10.121) results in

$$\begin{aligned}\frac{\psi\left(i\Delta x,(n+1)\Delta t\right)-\psi\left(\tilde{x},n\Delta t\right)}{\Delta t}&=\left[\left(1+\frac{\Delta t}{2}\left(\frac{\partial}{\partial t}-\overline{u}\frac{\partial}{\partial t}\right)\right)\left(\frac{\partial\psi}{\partial t}+\overline{u}\frac{\partial\psi}{\partial x}\right)\right]_i^n\\&\quad+O\left(\Delta t^2\right)+O\left(\frac{\Delta x^3}{\Delta t}\right)=0.\end{aligned} \tag{10.124}$$

While it may not appear obvious at first why (10.124) is written in this form, the reason is to illustrate that the first order in time term will be zero as it contains the Eulerian form of the advection equation; as we are considering the case of no forcing, this term is zero, and therefore the quadratic interpolation scheme is order Δt^2 and $\frac{\Delta x^3}{\Delta t}$, where the leading third-order spatial term is given by

$$-\frac{\Delta x^3}{6\Delta t}\left[\frac{\partial^3\psi}{\partial x^3}\right]_i^n W_2^3.$$

It is important that the behavior of this third-order term in space, that is divided by a first-order temporal term be examined to ensure that the scheme does indeed converge to the differential equation as $\Delta x\to 0$ and $\Delta t\to 0$. There are three situations that can occur with respect to both the spatial and temporal step sizes:

Case 1: The first situation that we consider is where the grid is refined such that the ratio $\frac{\Delta t}{\Delta x}\to 0$ as $\Delta t\to 0$ and $\Delta x\to 0$. Given this situation, then $p\to 0$ and by association the Courant number $\alpha\to\overline{u}\frac{\Delta t}{\Delta x}$. In this situation the W_2^3 term would tend to $-\overline{u}\frac{\Delta t}{\Delta x}$, then the term which appears to be the ratio of the third-order space to first-order time would be reduced to $O\left(\Delta x^2\right)$.

Case 2: The second possible situation is where the grid is refined such that the ratio $\frac{\Delta t}{\Delta x}$ tends to a constant a, as Δt and Δx trends to zero. This then implies that $\Delta t=a\Delta x$, and as such $\frac{\Delta x}{\Delta t}W_2^3\to aW_2^3\left(\alpha,\alpha-a\overline{u}\right)$, which is a constant, then again it appears that the term $O\left(\frac{\Delta x^3}{\Delta t}\right)$ is $O\left(\Delta x^2\right)$.

Case 3: The third and final possible situation with respect to the refining of the spacial and temporal grid is where $\frac{\Delta x}{\Delta t}\to 0$ as $\Delta t\to 0$ and $\Delta x\to 0$. Then we have that $|\overline{u}|\frac{\Delta t}{\Delta x}\gg|\alpha|$ and $p\to\overline{u}\frac{\Delta t}{\Delta x}$. In this case, we have that $W_2^3\to\left(\overline{u}\frac{\Delta t}{\Delta x}\right)^3$ and the term which appears to be ratio of the third order in

space to first order in time appears to be $O\left(\Delta t^2\right)$. Since the temporal step size is assumed to be much greater than the spatial step size in this case we have that the quadratic interpolation results in an $O\left(\Delta x^2\right)$ and $O\left(\Delta t^2\right)$.

Therefore, given the three scenarios described and analyzed above, we have shown that the quadratic Lagrange interpolation semi-Lagrangian approach is a $O\left(\Delta t^2\right)$ and $O\left(\Delta x^2\right)$ scheme. The analysis stated above comes from [281], where the motivation was to find a semi-Lagrangian scheme that had the desired rate of convergence just stated but also would be of a higher-order accuracy than the equivalent Eulerian approach.

Exercise 10.10. *Show that the cubic Lagrange interpolation-based semi-Lagrange scheme is consistent with the advection equation and show that it is equivalent to* $\mathcal{O}\left(\Delta x^4\right)$.

Exercise 10.11. *Determine the order of accuracy of the cubic Lagrange interpolation semi-Lagrangian scheme.*

Exercise 10.12. *Show that the central difference approximation for the derivative at the end points with the Hermite interpolation polynomial is consistent with the semi-Lagrangian advection differential equation.*

Exercise 10.13. *Determine the order of accuracy of the central difference-based Hermite interpolation.*

Exercise 10.14. *Show that the cubic spline interpolation provides a consistent approximation to the advection equation.*

Exercise 10.15. *Determine the order of accuracy of the cubic spline approximation to the Lagrangian advection equation.*

10.6 Semi-Lagrangian Schemes for Non-Constant Advection Velocity

In more advanced modeling of geophysical flows that contain a pure advection component to them, it is highly likely that the advection velocity $\overline{u}$ is not constant neither throughout the spatial domain nor for all time. This is important as we need to obtain a fairly accurate estimate of the departure point to ensure that the value there is the most accurate estimate for the tracer/field of interest at the arrival time. In this situation, the Eulerian form of the advection equation becomes

$$\frac{\partial \psi\left(x,t\right)}{\partial t}+u\left(x,t\right)\frac{\partial \psi}{\partial x}=0, \tag{10.125}$$

where $u\left(x,t\right)$ is a given function. **Note:** The field is still conserved following the parcel of fluid. This means that the semi-Lagrangian finite difference approach is still valid; but now the departure point is obtained by solving the equation

$$\frac{dx}{dt}=u\left(x,t\right). \tag{10.126}$$

This now raises an extra problem of solving (10.126). It is possible to write (10.126) as

$$x\left(t+\Delta t\right)-x\left(t\right)=\int_t^{t+\Delta t}u\left(x\left(\omega\right),\omega\right)d\omega, \tag{10.127}$$

where ω is a dummy time variable.

The expressions given in (10.127) is a highly implicit system that must be solved to obtain an estimate of the departure point, $x(t) = \tilde{x}$, assuming that the arrival point, $x(t + \Delta t)$, has a known location at $i\Delta x$.

We first consider the case where we assume the simplest possible approach, which is to assume that the advection velocity $u(x(\omega), \omega)$ is a constant $u_i^{n+\frac{1}{2}}$. Thus from (10.127) we have $\tilde{x} = i\Delta x - \Delta t u_i^{n+\frac{1}{2}}$. Substituting the new expression for the departure point into the expression for the Courant number yields $p + \alpha = u_i^{n+\frac{1}{2}} \frac{\Delta t}{\Delta x}$; next substituting this expression into Taylor series expansion of the tracer at the arrival point's equivalent at time $t = t_n$ assuming a quadratic Lagrange interpolation formula for the estimation of the value of the tracer at the departure point, which enables the value of the tracer at the departure point to be expanded as a Taylor series about $(i\Delta x, n\Delta t)$. Substituting all this information into the semi-Lagrangian forward upwind expression for the time derivative results in a scheme that is only first order, which is of no use.

Given this disappointing setback, we consider an approach suggested in [368], where an iterative approach could be used to solve the implicit relationship between the departure point and the advection velocity. The basic idea of the iterative approach is to try to obtain the best estimate of the advecting velocity at the center of the trajectory as possible and then approximate (10.127) as

$$x(t) = x(t + \Delta t) - \Delta t u\left(x\left(t + \frac{\Delta t}{2}\right), t + \frac{\Delta t}{2}\right),$$

which enables us to still obtain second-order accuracy. The estimation of $x\left(t + \frac{\Delta t}{2}\right)$ is taken as half of the position of the arrival point, plus the previous best estimate of the departure point position, i.e.

$$x^{(k)}\left(t + \frac{\Delta t}{2}\right) = \frac{1}{2}\left(x(t + \Delta t) + x^{(k)}(t)\right), \tag{10.128}$$

and thus we can approximate (10.127) to obtain the next estimate of $x^{(k)}$ as

$$x^{(k+1)}(t) = x(t + \Delta t) - \Delta t u\left(x^{(k)}\left(t + \frac{\Delta t}{2}\right), t + \frac{\Delta t}{2}\right). \tag{10.129}$$

This process is continued until we obtain convergence.

Once the scheme has converged and the best estimate of the departure point has been obtained, it is possible to evaluate the interpolation polynomial for a best estimate of the tracer at that point.

An interesting twist in the case of the iterative approach suggested in [368] and then expanded upon in [281], is the situation where it is possible to obtain the same second-order approximation to the advection equation. This accuracy arises from considering the first guess for the departure point, which initially is

$$\tilde{x}^{(0)} = x^{(0)}(t) = i\Delta x - \Delta t u_i^{n+\frac{1}{2}},$$

and then using (10.128) and (10.129), results in the first-order estimate of the departure point as

$$\tilde{x}^{(1)} = i\Delta x - \Delta t u\left(i\Delta x - \frac{\Delta t}{2} u_i^{n+\frac{1}{2}}, \left(n + \frac{1}{2}\right)\Delta t\right). \tag{10.130}$$

However, in the work by McDonald throughout the very well written papers from the 1980s and the early 1990s, he suggests perhaps using a linear Lagrange interpolating to compute the first guess of $x^{(1)}$ in (10.130) as

$$u\left(i\Delta x - u_i^{n+\frac{1}{2}}\frac{\Delta t}{2}, \left(n+\frac{1}{2}\right)\Delta t\right) = \left(1-\hat{\gamma}\right)u_{i-m}^{n+\frac{1}{2}} + \hat{\gamma}u_{i-m-1}^{n+\frac{1}{2}}, \tag{10.131}$$

where $\hat{\gamma} = -m + \frac{\Delta t u_i^{n+\frac{1}{2}}}{2\Delta x}$ and m is an integer that is chosen such that $0 < \hat{\gamma} < 1$. To see the derivation of the proof that this approach yields a second-order scheme we recommend reading [281].

It is becoming quite clear that as the advection problem becomes more complex, and closer to a real-life geophysical problem, there are more step required to ensure that the desired order of accuracy of the semi-Lagrangian scheme is maintained. So far we have only considered forward upwind approximations to the time derivative in the semi-Lagrangian problem, and the non-forcing case. In the next section we shall present different approaches to deal with the non-zero forcing terms along with how using different order approximations to the time derivative component helps with the determination of the non-forcing terms at the departure point and additionally techniques for better estimates of the departure point through higher-order time derivative approximations.

10.7 Semi-Lagrangian Scheme for Non-Zero Forcing

So far we have considered the case where we apply the semi-Lagrangian approach to a non-forcing term advective equation, $\frac{D\psi}{Dt} = 0$. We have considered both the constant advecting velocity and the non-constant advecting velocity where we needed to iterate between the estimates of the departure point and the advecting velocity.

When a forcing term is present, it has to be taken into account in the updating of the value of the tracer at the arrival point through the back trajectory.

As we saw when introducing the different forms of the finite difference approximations for the initial value problem in Chapter 8, we can consider both explicit and implicit approximations to the forcing term. Some approximations to the forcing term are referred to as **semi-implicit**, and the whole Lagrangian scheme applied as a **semi-implicit, semi-Lagrangian** method; this is where the forcing term is evaluated at both time $n+1$ and time n, as shown in (10.132);

$$\frac{\psi\left(x_i, t^{n+1}\right) - \psi\left(\tilde{x}_i, t^n\right)}{\Delta t} = \theta f\left(x_i, t^{n+1}\right) + (1-\theta) f\left(\tilde{x}_i, t^n\right), \tag{10.132}$$

where if $\theta = 1$, then we have the equivalent to an implicit Euler method, while if we took $\theta = \frac{1}{2}$, then we have the trapezoidal rule. Finally, if we set $\theta = 0$, then we have the explicit Euler method.

However, we have to determine the value for the forcing term at the departure point at time t^n, and as such we would use the same order of interpolation polynomial as that which is used for the Lagrange differential operator's numerical approximation for a constant advecting velocity.

Therefore, discretizing (10.132) with a simple forward upwind scheme for the time derivative results in the general update for ψ_i^{n+1} as

$$\psi_i^{n+1} = \tilde{\psi}_i^n + \Delta t f\left(\tilde{x}_i, t^n\right). \tag{10.133}$$

Semi-implicit, semi-Lagrangian approaches are have been used quite extensively in many of the world's operational numerical weather prediction models that suits their numerical needs. However, most of their situations would be for the non-constant advecting velocity, and with forcing terms that are functions of different dynamical scales. As an aside here, it should be noted that there is extensive research in the atmospheric data assimilation/modeling communities to change the underlying grids of the numerical models. The trend at this time is toward more polygon-based rather than grid- or spectral-based numerical models; we shall go into more detail about spherical model in Chapter 12.

Given this new form of the underlying numerical grid, some atmospheric numerical modeling centers and research groups are moving back toward the Eulerian approach for advection, in part due to the fact that the departure points are not required to be calculated relative to a node of the polygon. However, not all numerical weather prediction centers are heading that way, and for other forms of advection in geophysical models, that are not meteorological-based, the need for polyhedral grids may not be a priority.

Returning to the advection equation, if we consider a semi-implicit, semi-Lagrangian approach, then the associated discrete equation could be of the form

$$\frac{D\psi}{Dt} = \frac{1}{2} f\left(\tilde{x}_i, t^{n+1}\right) + \frac{1}{2} f\left(\tilde{x}, t^n\right).$$

The reason for considering the forcing terms as the sum of an implicit and an explicit component is motivated by separating the forcing term into a fast linear term and a slower term, which is referred to in [28] as a residual nonlinear term as

$$\frac{D\psi}{Dt} = \mathsf{N} + \mathsf{L}. \tag{10.134}$$

It is assumed that the two terms on the right-hand side of (10.134) depend on the tracer $\psi(x,t)$, and where it is assumed that we know the velocity at each time step. To solve (10.134), it is quite common for an explicit-based discretization to be used for the nonlinear term, while implicit schemes are used for the linear term. An interesting point about [28] is that different approximations to the semi-implicit semi-Lagrangian formulations is a cutting edge area of research, as more desirable properties for the schemes are sought.

We should note here that we also have to take into account the possibility that the advecting velocity is not constant in time and space, and as such will affect where the departure points are. This is an important point, as stated in [283]; even for a varying advecting velocity, it was quite common to take the velocity at the arrival point to find the back trajectory to the departure point.

In [283] the authors propose using a weighted average scheme between two levels to estimate the advecting velocity at the $n + \frac{1}{2}$ time level through

$$u\left(I\Delta x, \left(n+\frac{1}{2}\right)\Delta t\right) = \frac{3}{2} v\left(I\Delta x, n\Delta t\right) + \frac{1}{2} u\left(I\Delta x, (n-1)\Delta t\right). \tag{10.135}$$

Briefly returning to the non-forcing semi-Lagrangian methods, as shown in [411], it is possible to apply a second-order, time centered, approximation to the Lagrangian time derivative itself as

$$\frac{\psi\left(x_i, t_n + \Delta t\right) - \psi\left(x_i - 2\alpha_i, t_n - \Delta t\right)}{2\Delta t}, \tag{10.136}$$

which is an example of a three-level time scheme; where it is assumed that the back trajectory is integrated over two time steps, and that the parcel that we are following, travels the same distance in each time step. Below is a copy of the schematic for the three-level scheme from [411] illustrating the assumptions made.

From Fig. 10.11 it is stated in [411] that if we know α_i, then it is possible to find the value of the quantity at time $t + \Delta t$ through evaluating the upstream value of the quantity at $\psi\,(x_i - 2\alpha_i, t - \Delta t)$. Again we are assuming that the velocity does not change over the two time steps and as such the same distance from the arrival point location is the same over the two time steps. Therefore, we have the Roberts method, where to find α we have to iterate $\alpha_i = \Delta t u\,(x_i - \alpha_i, t_n)$, as presented in Section 10.6.

The steps involved in solving the Roberts' iterative approach to find the displacement and then the values of the fields and forcing functions as well as the advecting velocity at the departure point are as follows:

- **(i)** Solve $a_i^{(k+1)} = \Delta t \overline{u}\left(x_i - \alpha_i^{(k)}, t_n\right)$ iteratively for the displacement for all the mesh points x_i using some form of initial guess. A quite common first guess is the value of the displacement at the previous time and an interpolation formula.
- **(ii)** Evaluate ψ at the upstream point $x_i - 2\alpha_i$ at time $t_n - \Delta t$, again using an interpolation formula.
- **(iii)** Obtain the value of the tracer at the arrival time by evaluating (10.136).

As we have just mentioned this is equivalent to a three time step semi-Lagrangian method where we are using time t_{n-1}, t_n, and t_{n+1}, which was for the zero forcing case but where the advecting velocity is not constant.

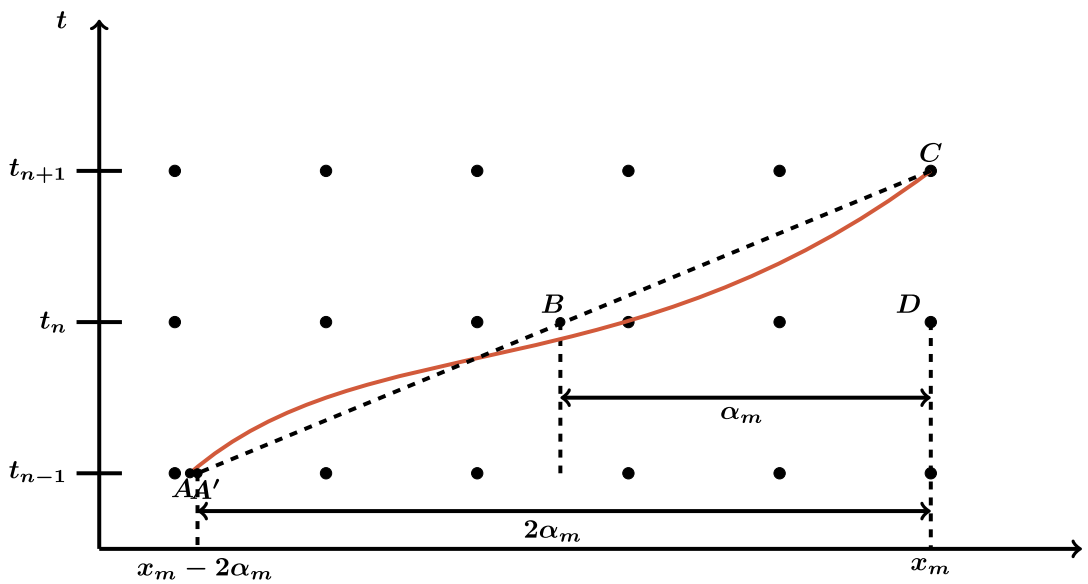


FIGURE 10.11

Recreation of the three time level advection stencil from Staniforth and Côté, (1991).

We now consider the scheme from [285], where although the formulation is for two-dimensional spherical coordinates, we shall go into numerical modeling on the sphere in Chapter 12, here we shall explain the theory for a one-dimensional Cartesian coordinate system.

We start by assuming that a dynamical quantity is integrated in a semi-Lagrangian form, implying that (10.134), when discretized in time, has the solution

$$\psi^{n+1} - \tilde{\psi}^n = \int_t^{t+\Delta t} \left(\mathsf{N}(x,t) + \mathsf{L}(x,t)\right) dt, \tag{10.137}$$

where $\tilde{\psi}$ is the value of ψ at the departure point. However, unlike before where we only searched for the departure point at the current time step, it is stated that for a variable advecting wind, $u(x,t)$, the departure point is calculated as $x - \hat{u}\Delta t$, where $\hat{u}$ is calculated as a space- and time-centered estimate along a trajectory.

The integral in (10.137) is along a trajectory of a parcel of air. For the linear term, it is then possible to use the following semi-Lagrangian estimate:

$$\int_t^{t+\Delta t} \mathsf{L}(x,t)\, dt = \left(\frac{\Delta t}{2}\right)\left(\mathsf{L}^{n+1} - \hat{\mathsf{L}}^n\right), \tag{10.138}$$

which is correct to $O\left(\Delta t^2\right)$.

However, it is not recommended to use the approach in (10.138) for the nonlinear term, as it leads to a complicated system of equations that is difficult to solve [285]. It is stated in [285] that if the nonlinear terms are small enough, then it is possible to integrate through an explicit approach without causing instability. As an example [285] introduces two different explicit schemes to approximate the integration of the nonlinear term. The first of these scheme is referred to as a *centered Lagrangian explicit (1) scheme*, which comes from [434] and is defined as

$$\int_t^{t+\Delta t} \mathsf{N}(x,t)\, dt = \Delta t \tilde{\mathsf{N}}_{\frac{1}{2}}^{n+\frac{1}{2}}, \tag{10.139}$$

where the nonlinear term is evaluated at the halfway point between the arrival point and the departure point such that this would be the value for the nonlinear term at $t + \frac{1}{2}\Delta t$, that is,

$$\tilde{\mathsf{N}}_{\frac{1}{2}}^{\frac{n+1}{2}} \equiv \mathsf{N}\left(x - \frac{u\hat{\Delta} t}{2}, t^{n+\frac{1}{2}}\right).$$

The second scheme which has the same order of accuracy as the first scheme, which is introduced in [285], and is referred to as the *centered Lagrangian explicit (2)* approximation is given by

$$\int_t^{t+\Delta t} \mathsf{N}(x,t)\, dt = \frac{\Delta t}{2}\left(\mathsf{N}^{n+\frac{1}{2}} + \tilde{\mathsf{N}}^{n+\frac{1}{2}}\right). \tag{10.140}$$

The value of the fields at the time level $n + \frac{1}{2}$ are computed through the second-order **Adams-Bashforth** scheme, which is given by

$$\psi^{n+\frac{1}{2}} = \frac{3}{2}\psi^n - \frac{1}{2}\psi^{n-1}. \tag{10.141}$$

Using the second-order Adams-Bashforth scheme means that the second centered scheme only requires one set of interpolation, that is,

$$\psi^{n+1} - \frac{\Delta t}{2}\mathsf{L}^{n+1} = \left[\psi^n + \frac{\Delta x}{2}\mathsf{L}^n + \frac{\Delta t}{2}\mathsf{N}^{n+\frac{1}{2}}\right]_D + \frac{\Delta t}{2}N^{n+\frac{1}{2}}, \tag{10.142}$$

where the subscript D refers to the departure point, whereas for the first centered scheme we have

$$\psi^{n+1} - \frac{\Delta t}{2}\mathsf{L}^{n+1} = \left[\psi^n + \frac{\Delta t}{2}\mathsf{L}^n\right]_D + \Delta t\mathsf{N}_{D/2}^{n+\frac{1}{2}}, \tag{10.143}$$

and therefore we require two sets of interpolations, the linear term to the departure point and then the nonlinear term to the halfway point on the trajectory from the departure point.

It can easily be shown that the relative accuracy of the two centered approaches, through using the equations in [283,285], are the same when a linear or a higher-order interpolation polynomial is used to estimate the departure point values for the nonlinear term for the various trajectories.

As mentioned at the beginning of this section, there is still a lot of work being undertaken to optimize semi-Lagrangian schemes. This chapter has served as an introduction but other areas that may be of interest to the reader but we shall not show here are the non-interpolating versions of semi-Lagrangian schemes.

One such scheme is to write the trajectory as the sum of two vectors, one of which goes to the nearest grid point to the departure point, while the other vector is the residual. The advection along the first trajectory is done in a Lagrangian form as we know the value of the function at this grid point, and therefore does not require any interpolation. For the second vector the advection is applied through an undamped three time level Eulerian approach such that the Courant number is less than one and we therefore maintain the stability properties of semi-Lagrangian schemes. The motivation for this approach was to remove the damping effect shown for the smaller scales with the interpolation approaches [367].

Returning to McDonald, where he has a very insightful paper from 1999 [282], in which he summarizes different iterative techniques for finding the departure point, highlighting that when they tried to apply the semi-implicit semi-Lagrangian scheme into a high-resolution meso-scale with a standard approach to find the departure point, then the associated 24 h forecast became incredibly noisy. McDonald inferred that this noise was coming from features that are normally too small relative to the synoptic scale modeling size grids.

While we have shown the semi-Lagrangian in finite difference formulation, it does also work with finite elements as well as with finite volume and spectral methods. There are also flux form semi-Lagrangian advection schemes. A more detailed description and applications of many more semi-Lagrangian methods in spherical coordinates, in 3D, with finite volumes, to name but a few more areas can be found in [131].

Given all the theory that we have presented in this chapter and over all of the Cartesian-based discretizations that we have shown, we shall look at an example of comparing the effects of different finite difference and semi-Lagrangian schemes with a linearized quasi-geostrophic potential vorticity (QGPV) model, also known as the Eady model.

10.8 Example: 2D Quasi-Geostrophic Potential Vorticity (Eady Model)

In this section we introduce a two-dimensional model, where the model is an x-z strip that is an approximation to the atmosphere circling the Earth. The purpose of introducing this model is that it enables us to illustrate a lot of the theory that has been developed over the last few chapters that have been associated with numerical modeling. In the next section we shall introduce quasi-geostrophic theory, which is important for both atmospheric and oceanic modeling. A nice feature of the Eady model is that it is a periodic domain, with initial value problems on its z boundaries, and requires the solution of a Laplace or a Poisson equation in the interior. However, as indicated in the periodic boundary conditions section in the last chapter, the problem is ill-posed and as such we shall address techniques to overcome this feature's effects on the numerical problem. We shall also compare Eulerian and semi-Lagrangian methods for the initial value problems on the boundaries and illustrate some performance problems that the stability analysis indicated for the Lagrange-based interpolation polynomials. We shall compare two different approaches that are used to compensate for the ill-posedness that was identified toward the end of the last chapter.

The two-dimensional Eady model [100] is the most simple theoretical model used to study baroclinic instability in the x-z plane. It is a linear quasi-geostrophic (GC) model that can be used to describe the vertical coupling between waves at the tropopause and the surface, for a meteorological setup. The Eady model equations support two types of what are referred to as **normal mode** solutions [119]: neutral modes that correspond to boundary waves with short wavelengths and unstable long waves, that grow or decay exponentially.

The two-dimensional Eady model is a simple linear model for quasi-geostrophic flows and it is derived from considering simplifications to what are referred to in the meteorological and oceanic communities as the **primitive equations**. A detailed derivation of the Eady model can be found in [130]. Here we shall present the sets of differential equations that we need to numerically approximate for the advection of buoyancy and quasi-geostrophic potential vorticity (QGPV), which are.

$$q' = \nabla_h^2 \psi + \frac{\partial}{\partial z}\left(\frac{f_0^2}{N^2}\frac{\partial \psi}{\partial z}\right), \tag{10.144a}$$

$$D_g b' = -wN^2, \tag{10.144b}$$

and are important in the various versions of the Eady model.

The Eady model is used to investigate how perturbations evolve with time from thermal wind balance. The perturbations that are of interest are believed to generate mid-latitude cyclones and anticyclones, in the x-z plane. As with most meteorological modeling there are a few more important assumptions to be made before we arrive at the final set of equations for the Eady model. These assumptions are as follows:

- There is no basic state velocity in the y-direction.
- The static stability parameter, N^2, is constant throughout the region of interest.
- The Coriolis force, f, is also constant throughout the region.
- The region is to be bounded below by the Earth's surface and above by a non-varying height.

The upper boundary is usually taken as the tropopause, which is approximately 10 km above the Earth's surface.

The equations can be written in a yet simpler form by scaling the model on to $\hat{z} \in [0, 1]$ and $\hat{x} \in [0, 1]$. Therefore, the equations that define the Eady model in two dimensions, and the set of equations that we shall consider different numerical approximations to in the first set of experiments are given by

$$\frac{\partial^2 \psi}{\partial \hat{x}^2} + \frac{\partial^2 \psi}{\partial \hat{z}^2} = 0 \quad \text{in } \Omega \tag{10.145}$$

$$\left(\frac{\partial}{\partial \hat{t}} + \hat{z}\frac{\partial}{\partial \hat{x}}\right)\frac{\partial \psi}{\partial \hat{z}} = \frac{\partial \psi}{\partial \hat{x}} \quad \text{on } \partial\Omega_1 \tag{10.146}$$

$$\psi\left(0, \hat{z}\right) = \psi\left(1, \hat{z}\right) \quad \text{on } \partial\Omega_2, \tag{10.147}$$

where $\partial\Omega_1$ and $\partial\Omega_2$ are the z and x boundaries, respectively, and Ω is the interior which here is the unit square.

10.8.1 Numerical Approximations for the Eady Model

For the partial differential equations in (10.145) and (10.146), it is clear that there are two forms of numerical modeling that need to be applied to solve this problem; the first of these is to the Laplace equation $\nabla^2 \psi = 0$ in the interior, while the second set of numerical approximation are for the advection equations for the buoyancy on the z boundaries. For the simple case where $q' = 0$, we assign initial conditions for the buoyancy on both boundaries.

Numerical approximation to $\nabla^2_{x,z}\psi = q'$ in the interior

To solve the Laplace equation in the interior we apply the standard five-point stencil presented in Chapter 9. The five-point stencil is a second-order approximation in both x and y, but for our problem here it is still second order in z as well. However, we have to address the fact that we cannot explicitly apply the five-point approximation to $\nabla_{x,z}$ on the two z boundaries. To overcome this shortfall we use the property that buoyancy is equivalent to the z derivative of the streamfunction on the z boundaries. If we fit a second-order central difference to the first-order z derivatives, employing ghost points outside of the z boundaries, then we can obtain expressions for the ghost points in terms of the interior points. Given the five-point stencil and the new expressions to complete the approximation to the Laplace equation on the z boundaries, where we have employed periodic boundary conditions for the x-direction; this results in a matrix-vector equation of the form $\mathbf{A}\boldsymbol{\psi} = \mathbf{b}$.

However, as shown at the end of Chapter 9, the matrix $\mathbf{A}$ that results from this discretization is singular. Therefore, we need to consider approaches that can still be consistent with the continuous equations but enable us to obtain a discrete approximation. As mentioned before, the continuous problem itself is ill-posed, but we can overcome this shortfall in the numerical approximation by introducing new constraints on the problem.

The first new constraint that we consider and use in the numerical experiments that we shall present soon is to assign a specific value to one point in the domain to the streamfunction as $\phi_{i,j} = \phi_0$, where ϕ_0 is an arbitrary constant. For simplicity we take $\phi_0 = 0$ and assign this constraint to $\psi_{1,1}$. The new resulting $\mathbf{A}$ matrix is now strictly diagonally dominant for at least one row and diagonally dominant for the remaining, and hence the new matrix is invertible.

A second approach that is considered to overcome the singularity of the matrix associate with the discretization is to add a small parameter ε to all of the diagonal entries to perturb the matrix away from the singularity. We have to be cautious when using this approach, as the condition number for a nearly

singular matrix is very large and as such when we perturb away from a nearly singular matrix then this could perturb the numerical model more than the magnitude of the added constant. However, it could also improve the conditioning of the matrix as a by-product.

The inverses of the two different $\mathbf{A}$ matrices, denoted $\mathbf{A_0}$ and $\mathbf{A_\varepsilon}$, are achieved through using an LU decomposition function in MATLAB®, which is solving $\mathbf{PA} = \mathbf{LU}$ and then creating the inverse of $\mathbf{A}^{-1}$ through $\mathbf{A}^{-1} = \mathbf{P} \setminus (\mathbf{L} \setminus \mathbf{U})$ where $\setminus$ is an inversion technique in MATLAB which uses a Gaussian elimination approach. We store the inverse matrices as they do not change throughout the time of the simulations and is faster than applying the Gaussian elimination technique at each time step.

10.8.2 Numerical Approximations to the Advection Equation

As we saw in the derivation of the Eady model, we have two different types of partial differential equations that we have to approximate; the first set of partial differential equations that we consider are the advection equations. As we have seen there are two quantities that are advected in the Eady model: the buoyancy on the z boundaries and the QGPV perturbation in the interior. However, we consider both the case where there is not perturbation to the QGPV, $q' = 0$, which implies that there is no advection equation to solve in interior, just on the two z boundaries, but we still have to solve for the streamfunction in the interior which is found through inverting a Laplace equation. The second case is where there is a perturbation to the QGPV, $q' \neq 0$, which implies that we have three advection equations to solve, but now we have to invert a Poisson equation in interior.

For the first case we have to numerically approximate following two partial differential equations:

$$\begin{aligned} &\frac{\partial b'}{\partial t} + \frac{\partial b'}{\partial x} = \frac{\partial \psi}{\partial x}, && z = 1, \\ &\frac{\partial b'}{\partial t} = \frac{\partial \psi}{\partial x}, && z = 0, \end{aligned}$$

where the advecting wind is z, and as such on the lower boundary where $z = 0$ we only have the time derivative term and the forcing. From inverting the Laplace equation in the interior we are able to find the meridional wind perturbation which is the forcing term for the buoyancy advection.

When we consider the second test case, which is the full dynamical-based two-dimensional model with $q' \neq 0$ in the interior, we have an advection equation for q', without a forcing term, given by

$$\frac{\partial q'}{\partial t} + \overline{u}\frac{\partial q'}{\partial x} = 0,$$

where $\overline{u} = \Lambda z$ and Λ represents a constant shear with respect to the z-direction to numerically approximate.

For the first set of numerical experiments, where only advection of the buoyancy is considered, we investigate three different numerical schemes to approximate these partial differential equations. The three numerical schemes are considered are: explicit forward upwind (EFU), centered time, centered space (CTCS), and thirdly a cubic explicit semi-Lagrangian scheme (CESL). We shall now briefly describe each technique and their properties to verify what we hope to observe in this first experiment.

Explicit forward upwind

The EFU numerical approximation is a first-order discretization scheme, which is given by

$$\frac{b_i^{k+1} - b_i^k}{\Delta t} + \frac{b_i^k - b_{i-1}^k}{\Delta x} = \frac{\psi_{i+1,j}^k - \psi_{i-1,j}^k}{2\Delta x}. \tag{10.148}$$

It can be easily be shown that the approximation above is consistent with the advection equation, given the techniques that have been introduced. Applying a Neumann stability analysis to this scheme shows that there is a stability region which ensures that the scheme remains stable. This stability region for this is

$$\left|1 - \nu + \nu \exp^{-il\Delta x}\right| \leq 1, \tag{10.149}$$

where $\nu = \frac{\Delta t}{\Delta x}$. The constraint on the ratio of the temporal step size to the spatial step size is $\frac{\Delta t}{\Delta x} \leq 1$.

Centered-time, centered-space scheme

The CTCS scheme is a second-order approximation to the advection equation in both time and space. However, at the first time step, $k = 0$, we cannot apply a central difference in time so we shall apply an explicit forward time centered space scheme. It is well known that this scheme is unconditionally unstable, and as such we minimize its use to one time step. If we wanted to avoid the possible instability associated with this scheme, than an alternative approach would be to use a Lax-Wendroff scheme, which would involve one implicit step but would still be stable for that initial time step.

The difference equation for the CTCS scheme is given by

$$\frac{b_i^{k+1} - b_i^{k-1}}{2\Delta t} + \frac{b_{i+1}^k - b_{i-1}^k}{2\Delta x} = \frac{\psi_{i+1,j}^k - \psi_{i-1,j}^k}{2\Delta x}. \tag{10.150}$$

The amplification factor for this scheme is

$$-j\nu \sin(l\Delta x) \pm \sqrt{-\nu^2 \sin(l\Delta x) + 1}. \tag{10.151}$$

For the scheme to be stable, we require that the region defined by (10.151) lie in or on the unit circle. For this to be the case the square root quantity must be complex $\forall l$. This implies that $\nu \leq 1$, which is the same as the condition for the EFU.

Cubic Lagrange interpolation semi-Lagrangian scheme

Given that this chapter has focused on semi-Lagrangian theory, it is only natural that we have a semi-Lagrangian scheme to compare to the two Eulerian-based methods just mentioned. We shall use the cubic Lagrange interpolation given by

$$\begin{aligned}\tilde{\psi}\left(\tilde{x}_j^n, t^n\right) = &\frac{\alpha\left(1-\alpha^2\right)}{6}\psi_{j-p-2}^n + \frac{\alpha(1+\alpha)(2-\alpha)}{2}\psi_{j-p-1}^n \\ &+ \frac{\left(1-\alpha^2\right)(2-\alpha)}{2}\psi_{j-p}^n - \frac{\alpha(1-\alpha)(2-\alpha)}{6}\psi_{j-p+1}^n.\end{aligned} \tag{10.152}$$

Recall that this scheme is unconditionally stable but does suffer from some damping effects for different wavelengths given the advecting speed which affects the rational component of the Courant number.

For the buoyancy advection equation, there is also a forcing term that needs to be evaluated at the departure point. We shall use an explicit formulation for the forcing term and as such use the same formula defined in (10.152) applied to the meridional wind forcing term, $\frac{\partial \psi}{\partial x}$, to find its value at the departure point. This then makes the difference equation for the advection for the buoyancy

$$b_{0,1}^{k+1}(x_i) = b_{0,1}\left(\tilde{x}_i, t^k\right) + \Delta t \psi_{x,0,1}\left(\tilde{x}_i, t^n\right).$$

We should note here that we also have a constant advecting velocity with respect to x and t, which makes the departure points easier to count.

10.8.3 Numerical Approximation to the Laplace Equation in the Interior

We shall apply the five-point stencil described in Chapter 9 to the streamfunction for this problem, which results in a matrix-vector equation of the form $\mathbf{A\Psi} = \mathbf{d}$ to invert. In the interior of the domain the entries of the $\mathbf{A}$ matrix are given by

$$\begin{aligned} A(i,i) &= -2\left(\Delta z^2 + \Delta x^2\right), \\ A(i \pm 1, i) &= \Delta z^2, \\ A(i, i \pm 1) &= \Delta x^2. \end{aligned}$$

The right-hand side of the matrix-vector equation associated with the five-point stencil, the vector $\mathbf{d}$, are all zero for the interior points of the domain.

We still need to solve the Laplace equation on the two z boundaries as we require the meridional winds, which are the x derivative of the streamfunction, for the forcing of the buoyancy advection. However, we cannot evaluate the z-direction central difference to the streamfunction on either z boundaries. To overcome this shortfall, we shall use ghost points to evaluate the fact that the buoyancy is the z derivative of the streamfunction.

If we consider the $z = 1$ boundary first then we have that

$$\begin{aligned} \frac{\psi_G - \psi_{i,j-1}}{2\Delta z} &= b_{1,i}, \\ \Rightarrow \psi_G &= \psi_{i,j-1} + 2\Delta z b_{1,i}. \end{aligned}$$

Therefore, substituting the expression for the ghost point into the five-point stencil results in the matrix entries associated with the z derivatives on the top boundary as $A(i, i-1) = 2\Delta x^2$. The remaining part of the definition for the ghost point goes over to the right-hand side of the matrix-vector equation into the $\mathbf{d}$ vector.

We can apply a similar argument for the ghost point associated with the z derivative on the lower boundary. This time the ghost point becomes $\psi_G \equiv \psi_{i,j+1} - 2\Delta z b_{0,i}$.

10.8.4 Buoyancy Advection on the Boundaries: $b_0' = 0$, $b_1' = \alpha \sin(K\Delta x)$

Due to the periodic condition on the buoyancy and the streamfunction in the x-direction, in this test problem we choose a sine wave of wavelength $K = 2n\pi$ and amplitude α, which is set to 20 for this test case.

The problem for this test case is defined as

$$\begin{aligned} \frac{\partial^2 \psi}{\partial z^2} + \frac{\partial \psi^2}{\partial x^2} &= 0 \quad \text{in } \Omega = (0,1) \times (0,1) \\ \frac{\partial \psi}{\partial z} &= \alpha \sin(K \Delta x), \quad \text{on } z = 1, \\ \frac{\partial \psi}{\partial z} &= 0, \quad \text{on } z = 0. \end{aligned} \tag{10.153}$$

From the equations presented in (10.153), it is possible to find the initial expression for the streamfunction. If we consider a solution of the form $\psi = \left(Ae^{Kz} + Be^{-Kz}\right)\sin(kx) + C$, where C is the constant of integration, at $t = 0$. From the extra condition to overcome the ill-posedness that $\psi(0,0) = 0$ then $C = 0$.

The expression for the initial streamfunction satisfies the interior differential equation in (10.153), as shown here:

$$\begin{aligned} \psi_x &= K\left(Ae^{Kz} + Be^{-Kz}\right)\cos(Kx), \\ \psi_{xx} &= -K^2\left(Ae^{Kz} + Be^{-Kz}\right)\sin(Kx), \\ \psi_z &= K\left(Ae^{Kz} - Be^{-Kz}\right)\sin(Kx), \\ \psi_{zz} &= K^2\left(Ae^{Kz} + Be^{-Kz}\right)\sin(Kx), \end{aligned}$$

and therefore

$$\frac{\partial^2 \psi}{\partial z^2} = -\frac{\partial^2 \psi}{\partial x^2}.$$

From the initial conditions for the buoyancy on the two z boundaries, relating them to the z derivative of the streamfunction, $b' = \frac{\partial \psi}{\partial z}$, the two constants A and B can be found by rearranging these conditions as

$$\frac{\partial \psi}{\partial z}(x,0) = 0 \Rightarrow A - B = 0, \tag{10.154}$$

$$\frac{\partial \psi}{\partial z}(x,1) = \alpha \sin(Kx) \Rightarrow K\left(Ae^{K} - Be^{-K}\right) = \alpha. \tag{10.155}$$

From (10.154) we obtain the condition that

$$A = B. \tag{10.156}$$

Substituting (10.156) into (10.155) results in

$$\begin{aligned} KA\left(e^{K} - e^{-K}\right) &= \alpha, \\ \Rightarrow A &= \frac{\alpha}{2K\sinh(K)}. \end{aligned} \tag{10.157}$$

Hence the final form for the initial streamfunction is

$$\psi(x,z,0) = \frac{\alpha}{K\sinh(K)}\cosh(Kz)\sin(Kx). \tag{10.158}$$

In the numerical experiments we present results from, there are 64 grid points in the x-direction, with 8 vertical levels in the z-direction. This then gives a mesh containing 512 grid points. To ensure stability of the two Eulerian schemes, we use a time step, $\Delta t = \frac{\Delta x}{10}$. The model is run for 640 time steps.

In Figs. 10.12–10.14 we have plotted the solutions for the three numerical approximation to the buoyancy advection on the z boundaries, in the order that they have been introduced. We have plotted the buoyancy at five different times between the initial time and $T = 640$ time steps: the initial time, $T/4 \equiv 160$ time steps, $T/2 \equiv 320$ time steps, $3T/4 \equiv 480$ time steps, and $T - 1 \equiv 639$ time steps, which is equivalent to 1 day, to illustrate how the wave is moving but also to highlight any effects the numerical schemes are having on the shape of the buoyancy.

In Fig. 10.12 we have the results from the EFU scheme; we can see that the buoyancy wave on the upper boundary is being advected. However, it can clearly be seen that the buoyancy is being damped by the numerical scheme. The damping becomes obvious when Fig. 10.12 is compared to the solution for the ECTCS in Fig. 10.13. An important feature that can be inferred from the plots of the upper and lower boundary buoyancy is that the numerical model in the interior is communicating between the two boundaries. Recall that the initial condition for the lower boundary was $b' = 0$.

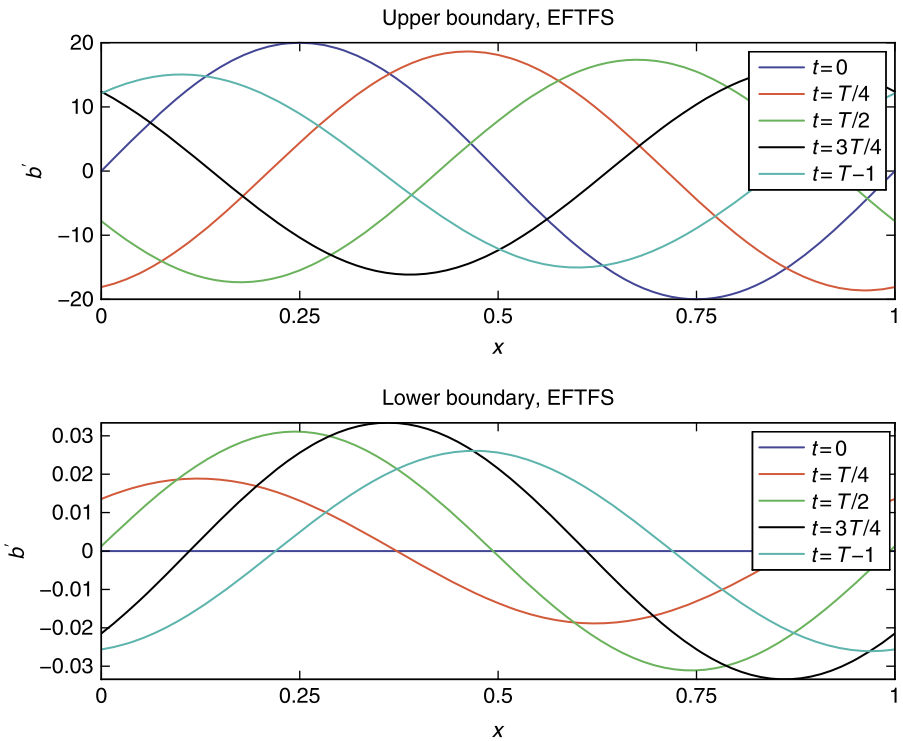


FIGURE 10.12

Plot of the buoyancy advection by the explicit forward upwind scheme.

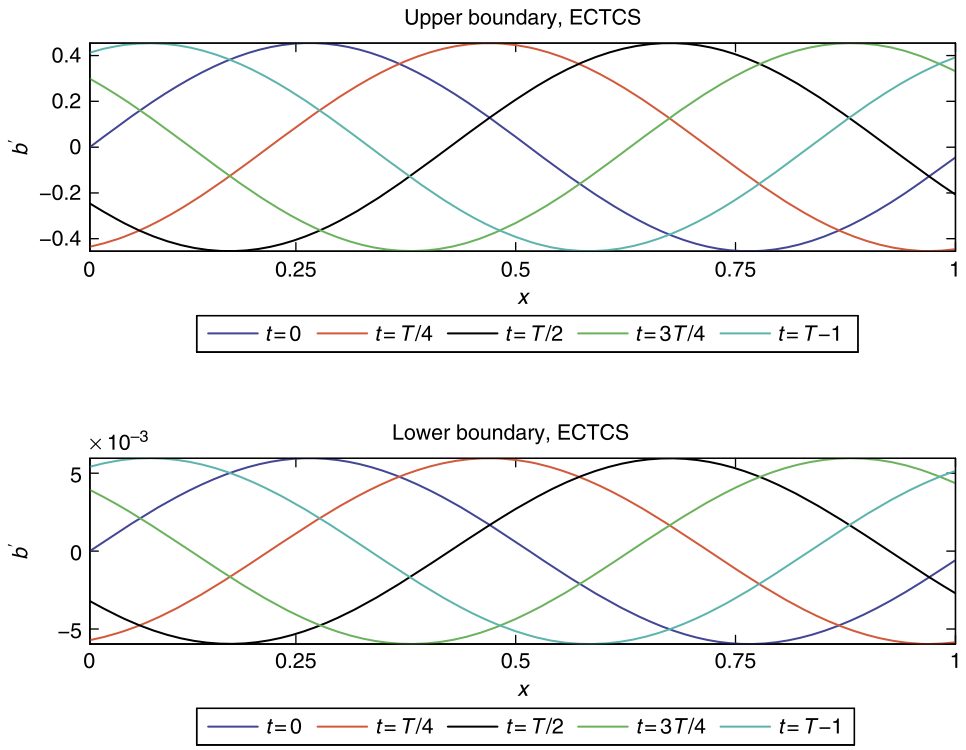


FIGURE 10.13

Plot of the buoyancy advection by the explicit centered-time centered-space scheme.

When comparing the solutions on the lower boundary between the EFU and the ECTCS, there are subtle differences in both the amplitude and location of the buoyancy wave. Therefore, the first-order scheme appears to be suffering from an artificial damping effect, which is consistent with the analysis of the amplification factor for this scheme. Finally in Fig. 10.14 we have plotted the solution from using the CESL scheme for the buoyancy advection. It is clear that the solution from the CESL is not suffering from the damping effect that the EFU scheme does. The results from the CESL and the ECTCS are almost identical.

10.8.5 Conditioning

The way in which the Eady model is defined for the continuous case results in an ill-posed problem, which in turns leads to a very ill-conditioned numerical problem. Earlier in this section we mentioned that there were two techniques that we would consider to overcome the ill-conditioning of the numerical problem: prescribe a set value to the streamfunction at a single point, or to add a small perturbation to all of the diagonal entries.

In [127] a study was performed to investigate the affect the two approaches just described have on the difference between the two associated solutions. It was shown that there was only a 10^{-7} differ-

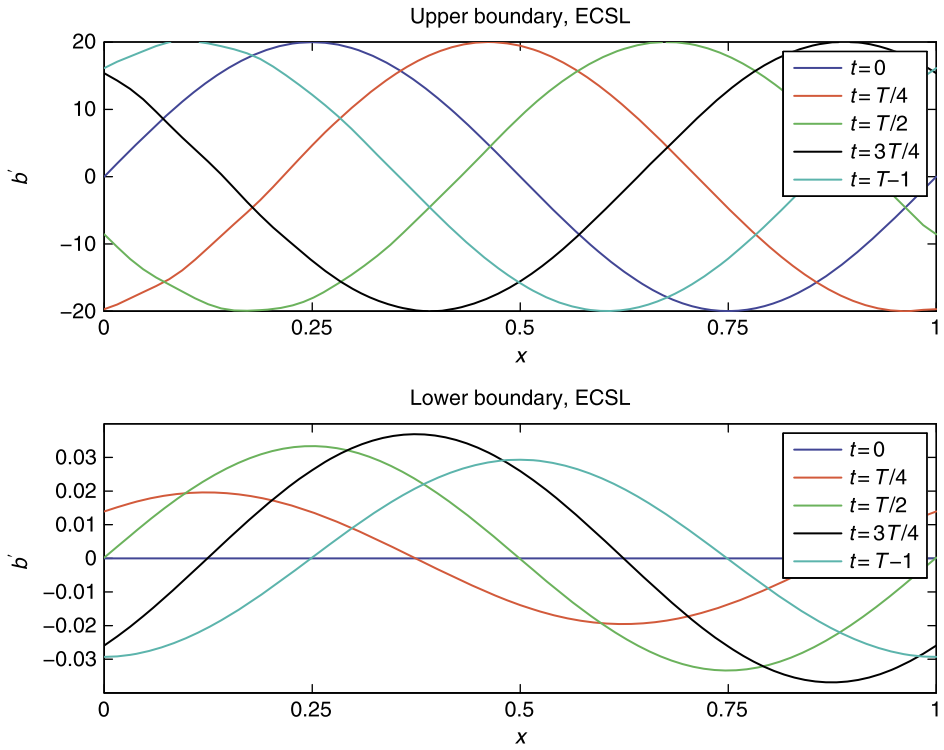


FIGURE 10.14

Plot of the buoyancy advection by the cubic semi-Lagrange scheme.

ence between the two solution when the grid was $N = 80$ with $J = 11$, for the solutions at the final time. However, Table 10.2 illustrates the effect that the small perturbation, $\varepsilon = 5 \times 10^{-15}$ had on the condition number, κ, compared to the preassigned value's equivalent **A** matrix.

An interesting feature to note about the condition numbers in Table 10.2 is that as the resolution of the interior mesh increases, the condition number of the preassigned **A** matrix is increasing. This is possibly due to the increasing number of zeros in the matrix as the matrix becomes more sparse, but for

Table 10.2 Condition Numbers κ, for $\mathbf{A}_0$ and $\mathbf{A}_\varepsilon$ Matrices, Where $\varepsilon = 5 \times 10^{-15}$.

N	J	κ ($\mathbf{A}_0$)	κ ($\mathbf{A}_\varepsilon$)
16	2	6.12×10^2	9.22×10^{14}
32	4	7.59×10^3	9.31×10^{13}
64	8	2.10×10^5	1.69×10^{13}
80	11	7.38×10^5	8.31×10^{12}

the perturbed case the condition number is decreasing; again this could be due to the increasing number of strictly diagonally dominant rows.

10.8.6 QGPV $\neq 0$

We now more on to the more advance numerical model where the initial perturbations that we use for this experiments are based upon those from [20]. The initial perturbation is for the meridional wind $v' = \frac{\partial \psi'}{\partial x}$. This perturbation is defined in the form $v' = V F_x(x) G(z)$, where $F_x(x)$ and $G(z)$ are defined as

$$F_x(x) = \begin{cases} -\sin^2(Kx) & -\frac{L}{2} \leq x < -\frac{L}{4}, \\ \sin(Kx) & -\frac{L}{4} \leq x < \frac{L}{4}, \\ \sin^2(Kx) & \frac{L}{4} \leq x < \frac{L}{2}, \\ 0 & \text{elsewhere}, \end{cases} \tag{10.159}$$

and

$$G(z) = \begin{cases} \cos^2(m(z - z_{\max})) & -\frac{D}{2} \leq z - z_{\max} < \frac{D}{2}, \\ 0 & \text{elsewhere}, \end{cases} \tag{10.160}$$

respectively, where L denotes the length of the perturbation in the horizontal direction. For the experiments L was taken to be 4000 km where now the domain is no longer the unit square but a realistic approximation in scale to a x-z plane in the atmosphere. The horizontal length scale is taken to be 16,000 km and the vertical length scale is 10 km. It is as a result of this scaling that the Coriolis parameter f_0 and the static stability parameter N have to be included. $z_{\max}$ represents where the center of the disturbance is in the z-direction and D is the vertical length scale of the disturbance, here taken to be 4 km. V is constant and for this example is set to 10.

We now use the property that the meridional wind is the x derivative of the streamfunction, which enables us to find a mathematical expression for the initial streamfunction perturbation through integrating $F(x)$ above. Thus the associated initial streamfunction perturbation, ψ', is defined by

$$\psi(x,z) = \begin{cases} \left(-\frac{L}{4} - \frac{x}{2} + \frac{\sin(2Kx)}{4K}\right)\cos^2(m(z - z_{\max})) & -\frac{L}{2} \leq x < -\frac{L}{4}, \\ & -\frac{D}{2} \leq z - z_{\max} < \frac{D}{2}, \\ \left(-\frac{L}{8} - \frac{\cos(Kx)}{K}\right)\cos^2(m(z - z_{\max})) & -\frac{L}{4} \leq x < \frac{L}{4}, \\ & -\frac{D}{2} \leq z - z_{\max} < \frac{D}{2}, \\ 0 & \text{elsewhere}. \end{cases} \tag{10.161}$$

Finally, given the expressions for the initial perturbations to the meridional winds and the streamfunction it is possible to derive the mathematical expression for the QGPV by taking the second derivative of $F(x)$ with respect to x, and the second z derivative of $G(z)$. This results in

$$F_{xx}(x) = \begin{cases} -2K\sin(Kx)\cos(Kx) & -\frac{L}{2} \leq x < -\frac{L}{4}, \\ K\cos(Kx) & -\frac{L}{4} \leq x < \frac{L}{4}, \\ 2K\sin(Kx)\cos(Kx) & \frac{L}{4} \leq x < \frac{L}{2}, \\ 0 & \text{elsewhere}, \end{cases} \tag{10.162}$$

$$G_{zz}(z) = \begin{cases} -2m^2\left(\sin^2(m(z-z_{\max})) + \cos^2(m(z-z_{\max}))\right) & -\frac{D}{2} \le z - z_{\max} < \frac{D}{2}, \\ 0 & \text{elsewhere.} \end{cases} \tag{10.163}$$

Therefore, combining (10.163) and (10.162) with the $F(x)$ component of (10.161), the initial perturbation to the QGPV can be defined by

$$q' = \left(F_{xx}(x) G(z) + F(x) G_{zz}(z)\right) V.$$

The initial perturbations to the streamfunction, meridional wind, and the QGPV defined above are presented in Fig. 10.15. For the streamfunction we can see that the perturbation is located in the center of the domain. The perturbations for the meridional wind are either side of the center of the domain and are of opposite sign. Finally, the QGPV perturbations are a dipole of negative, positive, then negative potential vorticity.

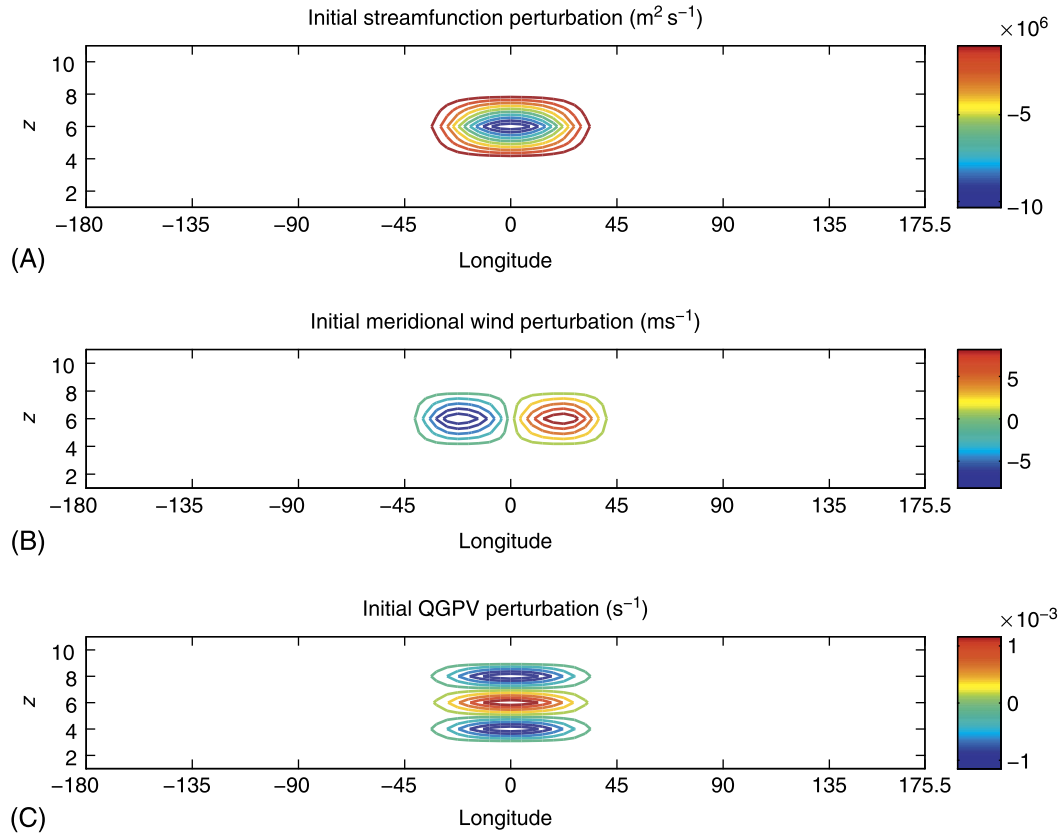


FIGURE 10.15

Plot of initial perturbations to (A) the meridional winds (v'), (B) the stream function (ψ'), and (C) the quasi-geostrophic potential vorticity (q').

The continuous equations that we are numerically approximating in this section are (10.144a) in the interior of the domain and (10.144b) for the buoyancy advection on the z boundaries. For the parameters required for (10.144a) we take $\overline{U}$ to be in the form of $\overline{U} = \Lambda z - u_0$, where $\Lambda = 4 \times 10^{-4}\ \mathrm{s}^{-1}$ and $u_0 = 20\ \mathrm{ms}^{-1}$. These choices for the parameters in $\overline{U}$ result in a linear shear with respect to the z-direction that starts at $-20\ \mathrm{ms}^{-1}$ on the lower z boundary and increases linearly to $20\ \mathrm{ms}^{-1}$ on the upper z boundary. A is taken to be equal to Λf_0 where $f_0 = 1 \times 10^{-4}$. Finally, we set the static stability parameter N to be $N = 1 \times 10^{-2}$.

The QGPV in the interior is advected by the ECTCS and the ECSL numerical approximations to (10.144b) in the full model, but we shall show results from the linear and quadratic Lagrange interpolation to show their damping effects on the QGPV.

In the vertical direction we use a grid spacing of 1 km, resulting in 11 levels, and in the horizontal direction we use a grid spacing of 200 km which results in 80 grid points. We use $\Delta t = 0.05$ days, which is equivalent to 4320 s. In the definitions for the meridional wind perturbations we set $L = 2000$ km, which makes $k = \frac{2\pi}{L}$, we set $D = 4$ km, so that $m = \frac{\pi}{D}$ and $z_{\max} = 5$ km. A more detailed meteorological justification for this set up can be found in [20]. We are interested in the numerical modeling component here and the effects that different numerical schemes have on the accuracy of the shape-conserving of the advection of the QGPV increment. A final comment about the setting up of this problem: we are using the constraint approach where we set $\psi_{1,1} = 0$ for the duration of the numerical experiment.

We are considering two set of experiments for the model setup. The first set of experiments involves comparing the advection of the potential vorticity for a long integration of 45 days, which is equivalent to 800 time steps. One of the reasons for the long integrations is to show that the numerical schemes can wrap around through the periodic boundary conditions. For this long integration run we compare the performance of the linear, quadratic, and cubic Lagrange interpolation polynomial-based advection as well as the explicit CTCS scheme.

The results from the long integration for the semi-Lagrangian scheme with the linear Lagrange polynomial are presented in Fig. 10.16. We can clearly see the damping effects that the linear scheme has, which were identified earlier in the stability analysis. For the model setup that we are considering here, the Courant numbers for each level are presented in Table 10.3.

We have swapped the direction of the interpolation schemes for levels where we have a negative advecting velocity to ensure that the Lagrange interpolation polynomial-based schemes are always upwind of the departure point.

The damping effects on the two negative PV balls is quite severe, such that the balls have lost over an order of magnitude. There is no diffusion in this model setup so the schemes should be as close to shape conserving as possible. These results indicate that the linear interpolation approach should not be used for these scales involved in the Eady model.

We now consider the results from the quadratic Lagrange interpolation, at the same 5-day intervals as the linear case are presented in Fig. 10.17. As indicated in the stability analysis for this scheme, there is still some damping occurring, but not as severe as for the linear case; however, we can see that some spurious detachments from the main PV negative balls does occur later on into the integration.

In Fig. 10.18 we present the results for the same set up as for the linear and quadratic semi-Lagrangian schemes but now for the explicit CTCS. We can see that the negative PV balls are being stretched in the direction which they are being advected, which again should not be occurring. This stretching, and the spurious detachment seen for the quadratic semi-Lagrangian case, are examples

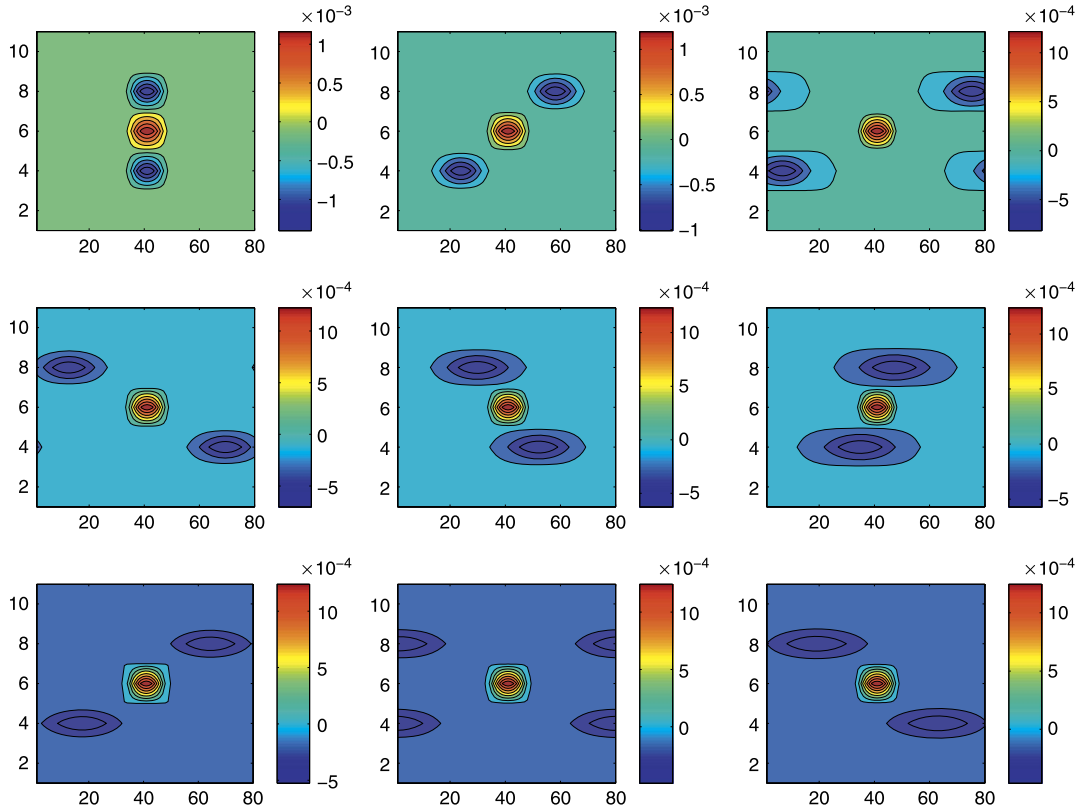


FIGURE 10.16

Plot of the long-term integration for the linear semi-Lagrangian scheme.

Table 10.3 Courant Numbers for the Different Vertical Levels Associated With the Different Advecting Wind Speeds.

Level	0	1	2	3	4	5	6	7	8	9	10
α	0.42	0.35	0.26	0.17	0.09	0	0.09	0.17	0.26	0.35	0.42

of the effects of **model error**, which plays an important role in the performance of data assimilation systems. We shall go into more detail about model error in Section 16.5.

In the plots for the ECTCS scheme at day 20 we start to see a spurious area of positive potential vorticity cutting off downwind of the negative potential vorticity balls. We recall that q' is the right-hand side term of the Poisson equation that we have to solve in the interior and on the boundaries to obtain the streamfunction, and hence by association the buoyancy perturbations as well. Ten days later there appears to be a small spurious ball of negative potential vorticity that is separated from behind the main ball which is not physical.

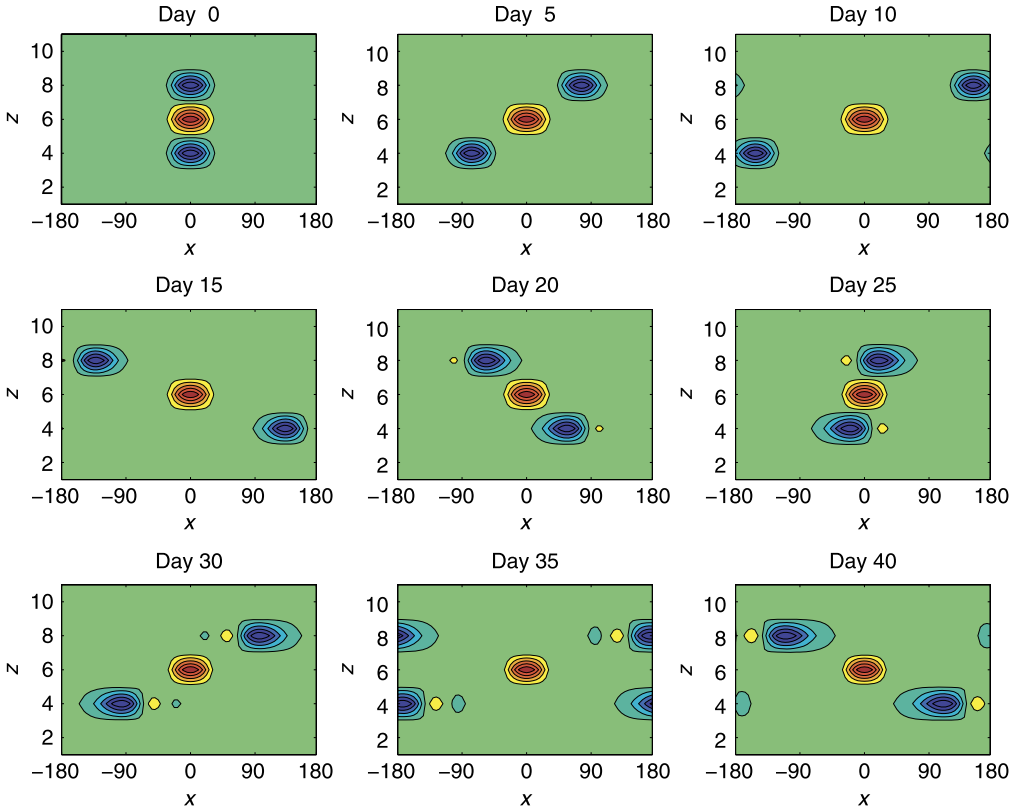


FIGURE 10.17

Plot of the long-term integration for the quadratic semi-Lagrangian scheme.

Finally moving onto the cubic Lagrange interpolation semi-Lagrangian scheme, Fig. 10.19, we can see that the distortion of the negative potential vorticity ball that is observed with the ECTCS scheme and to some extent with the quadratic semi-Lagrangian scheme, is not as pronounce with the cubic Lagrange interpolation. This is indicating that the cubic interpolation-based semi-Lagrangian approach is more shape conserving that if we use the ECTCS scheme. While there are more computations involved with the semi-Lagrangian approaches, we still have the advantage that semi-Lagrange schemes can use any time step for stability, but there are still restrictions on the time step for the accuracy. Therefore, to reduce computation time we could take larger time steps and then perform fewer evaluations of the scheme. Just as a reminder, we have used the same time step for all of the numerical schemes. Halving the time step size did not eliminate the spurious areas of QGPV in the model.

However, it should be noted that while it might appear that the cubic semi-Lagrangian approach may appear to be shape conserving, we have plotted the value of the PV for the $z = 4$ level which cuts straight through the lower negative PV ball for all four schemes for the first 5 days, 24 h apart, of integration to illustrate the severity of the overshoots in Fig. 10.20.

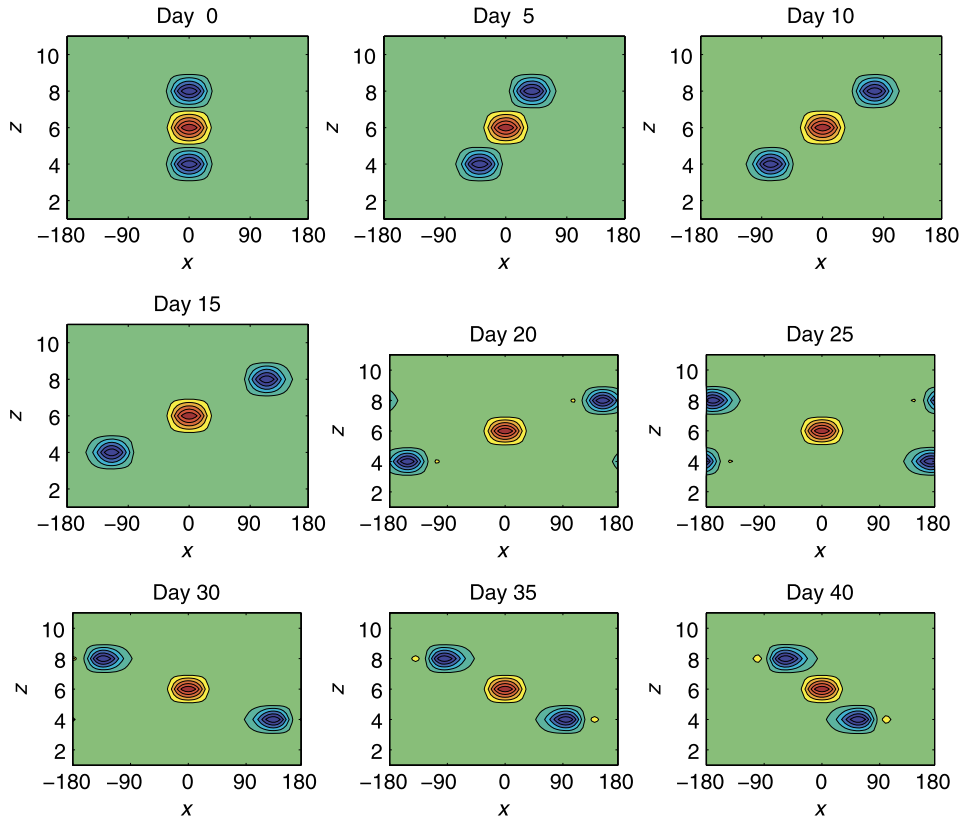


FIGURE 10.18

Plot of the long-term integration for the explicit centered-time centered-space scheme.

We can see for the linear Lagrange interpolation that the overshoots and the damping occur quite quickly and are quite large. For the quadratic and the ECTCS space we can see quite large overshoots, which are consistent with the stretching of the PV ball, but that the scheme appears to be preserving most of the mass. Finally, for the cubic Lagrange-based scheme we see that there are slight overshoots, but not to the extent as seen with the other three schemes (Fig. 10.21).

The final sets of plots that we present in this chapter are of the streamfunction at 24, 48, 72, and 96 h for the ECTCS and the cubic semi-Lagrangian scheme. These results are presented in Fig. 10.22. We can see from the plots that there are quite significant differences between the two solutions, recalling that the QGPV is the right-hand side of the Poisson equation that we have to invert to obtain new values for the streamfunction. We can see that there are differences in the slanting of the streamfunctions as well as the magnitude. The process that is going on in the interior is referred to **PV shielding** and we recommend reading [20] for a good description of the effects of the shielding. As the ECTCS scheme is stretching the PV, it is extending the area that the PV covers and as such prevents the streamfunction

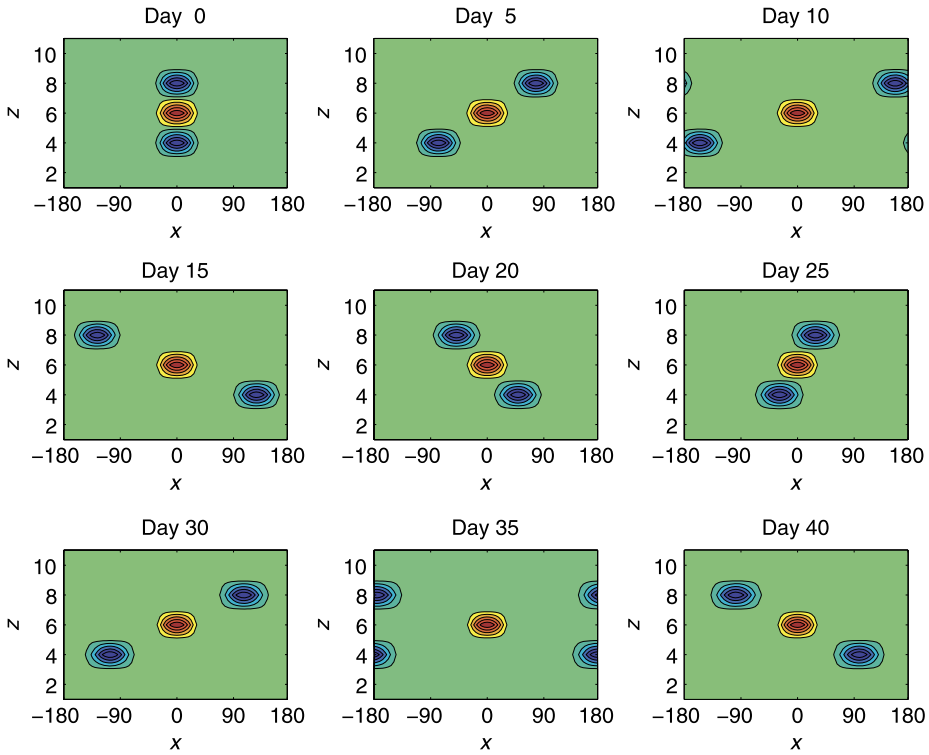


FIGURE 10.19

Plot of the long-term integration for the cubic semi-Lagrangian scheme.

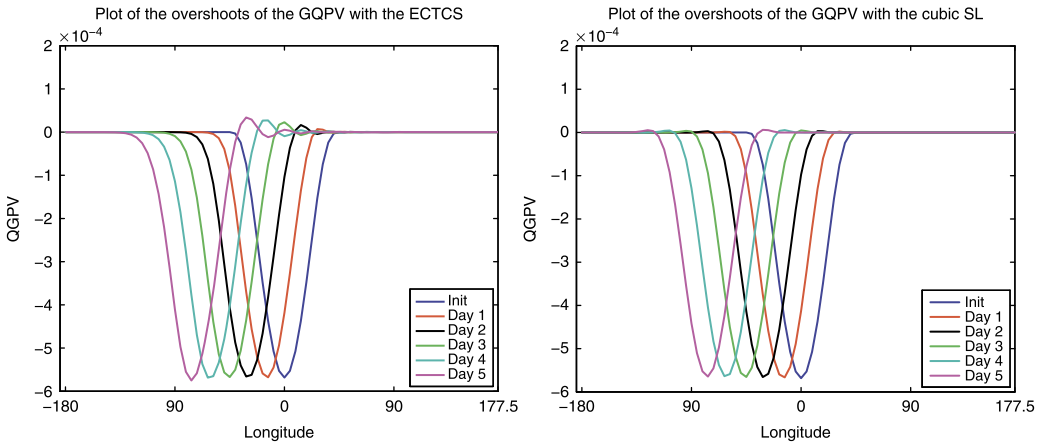


FIGURE 10.20

Plot of the overshoots for the different schemes for the first 5 days of integration 24 h apart for the ECTCS and the cubic semi-Lagrangian methods.

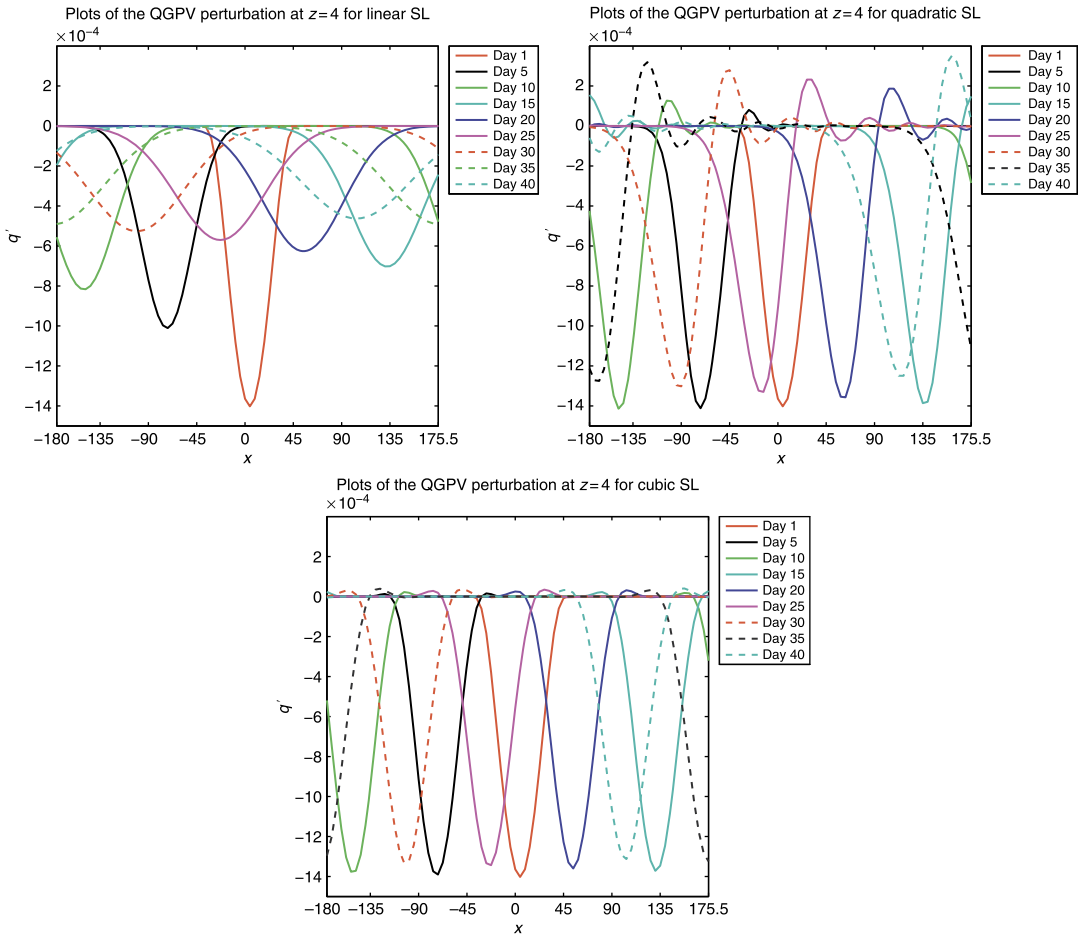


FIGURE 10.21

Plot of the overshoots for the different schemes for the first 5 days of integration 24 h apart.

from growing into this area. This feature is again an example of model error, as this extra shielding is an effect of the numerical scheme and not a physical process.

10.9 Summary

In this section we have introduced semi-Lagrangian-based numerical advection modeling schemes. We have derived different-order polynomials using the different interpolation formulas. These polynomial approximation were the Lagrange interpolation, the Newton divided differences, the Hermite polynomial, and the cubic splines. We have seen that when we have departure points that do not coincide

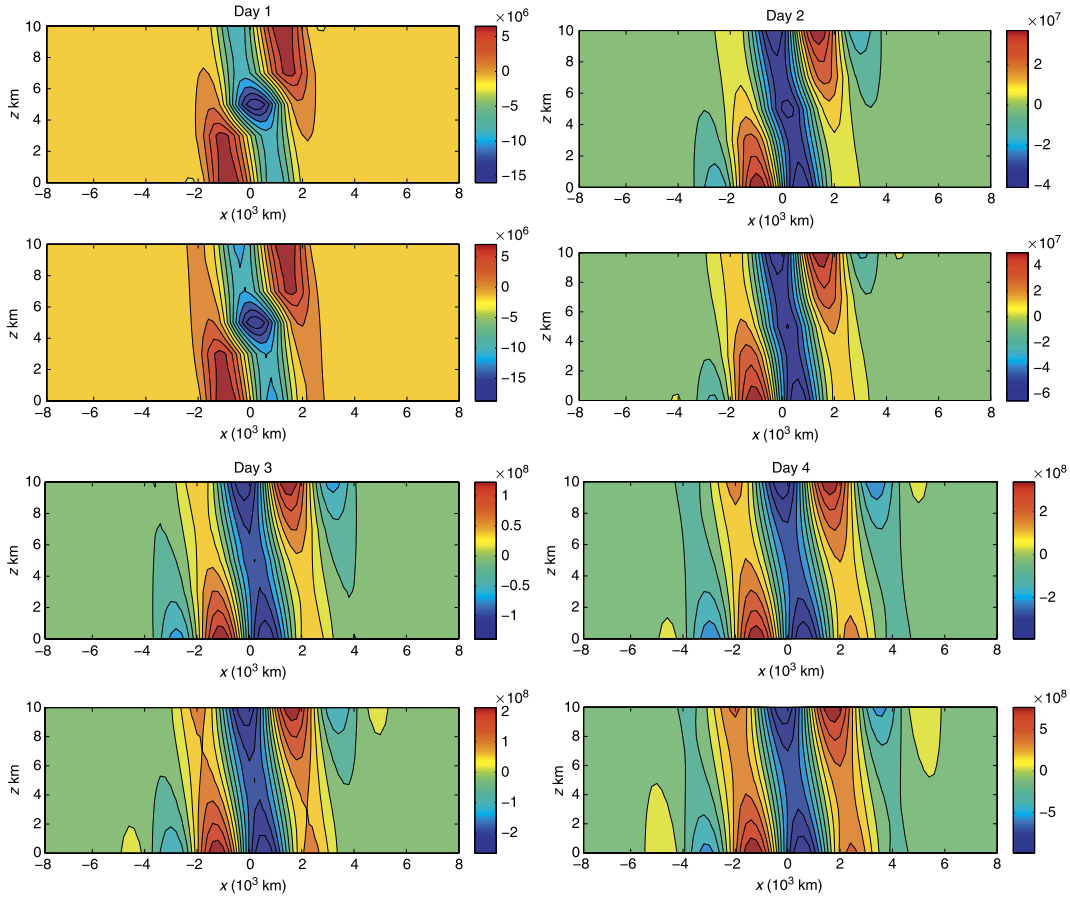


FIGURE 10.22

Plot of the evolution of the streamfunction perturbations for the explicit centered-time centered-space and the cubic semi-Lagrangian scheme at 24, 48, 72, and 96 h.

with the grid points, we split the Courant number into an integer component which corresponds to the number of grid points that the departure point is shifted from the arrival point, and a second component which was a rational component that corresponds to the percentage away from the nearest grid point the departure point was. Because of this representation of the Courant number, we saw that all of the schemes that we considered were unconditionally stable.

We have shown that the linear schemes quite heavily damp the lower wavelengths features, to the point that if the Courant number is 0.5, then the $2\Delta x$ feature is damped within one time step of this size. This effect was true of all of the interpolation schemes. However, when we have features that are larger than $2\Delta x$, we see that all of the schemes start to improve, but the higher-order cubic-based interpolation schemes had the least damping effect.

An important feature of the unconditional stable property of the semi-Lagrangian schemes is that it enables the user to take a larger time step than compared to the Eulerian explicit schemes, but we have to weigh up the time needed to determine the departure point, the Courant number, and finally interpolate the tracer to the departure point. We have also introduced the monotonicity conditions for the Hermite-based semi-Lagrangian schemes.

We have presented the theory for when we have varying advecting velocity which results in either an implicit iterative scheme involving the distance from the nearest grid point and the advecting velocity, or we can use finite difference approximations to estimate the velocity at a halfway time step. We have also introduced the semi-implicit semi-Lagrangian theory for when we have to solve the advection equation with a non-trivial forcing term which contains both nonlinear terms which we use an explicit scheme to solve, and a linear term that is solved through an implicit scheme.

Finally, in this chapter we have put a lot of the numerical model theory into practice with the linearized QGPV, or Eady, model. In this numerical example we have to solve either a Laplace equation, that is, $q' = 0$, or the Poisson equation for the case when $q' \neq 0$ in the interior of a x-z strip. We verified the damping effect of the lower-order semi-Lagrangian schemes as well as the lower-order explicit Eulerian-based schemes with either the buoyancy advection or the QGPV.

The purpose of this chapter was to consolidate as much as possible the information about semi-Lagrangian methods to highlight some of their pros and cons for use in any geophysical modeling where advection occurs, and also to highlight the effects that different choices for the interpolation polynomial have on the accuracy of the solution, which may have to be compensated for in a data assimilation scheme. Since the first edition the author has written a textbook going into a lot more details about semi-Lagrangian methods in [131].

We now move on to consider one more form of numerical modeling: **Finite element modeling**.

This page intentionally left blank

CHAPTER

Introduction to Finite Element Modeling

11

Contents

Finite element modeling, often abbreviated to FEM, is quite different compared to the different forms of discretizations presented so far. In this chapter we shall consider different methods from finite element theory to enable the reader to have a basic understanding of what is involved in FEM.

11.1 Solving the Boundary Value Problem

We start by defining the general problem, referred to as **problem A**, which is to find the function $u(x, y)$, such that it is the solution of the general elliptical partial differential equation:

$$-\nabla^2 u + qu \equiv -\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} + qu = f(x, y), \tag{11.1}$$

where as we saw in Chapter 9 there are three types of boundary conditions: Dirichlet, Neumann, or a mixture of the two. If we consider the domain in Fig. 11.1 we could have the Dirichlet condition $u = g(x, y)$ on the S_1 part of the boundary and $\nabla u \cdot \mathbf{n} = \frac{\partial u}{\partial \mathbf{n}} = \beta$ on S_2, or a combination $\frac{\partial u}{\partial \mathbf{n}} + \alpha u = \beta$, where $\mathbf{n}$ is the unit outward normal, with $q, \alpha \geq 0$ and g, f, q, α, β are given functions of x and y.

As mentioned above, the partial differential equation in (11.1) is an elliptical partial differential equation and this type of partial differential equation can represent:

- deflection of a membrane;

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00015-2

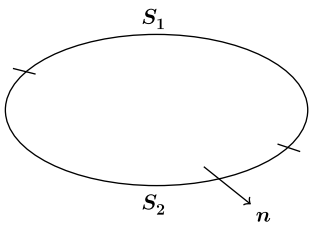


FIGURE 11.1

Diagram of a possible domain and its boundary for problem A.

- steady-state heat conduction;
- steady-state ground water flow;
- stress function in torsion; and
- rotational balanced flow.

The idea of finite elements is to approximate the domain R in Fig. 11.1 by using triangles, squares, or quadrilaterals. For the triangular approximation we would take $u \approx a + bx + cy$. This then leads to our first theorem for FEM, which is as follows.

Theorem 11.1. *Solving the general boundary value partial differential equation in* (11.1) *is equivalent to minimizing a functional of the form*

$$I[v] = \iint_R \left(\left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 + qv^2 - 2fv \right) dR + \int_{S_2} \left(\alpha v^2 - 2\beta v \right) ds, \tag{11.2}$$

over all $v(x, y)$ *such that:*

1. $V = g(x, y)$ *on* S_1*, this is the* ***essential boundary condition****; and*
2. *it is possible to obtain* $I[v]$.

In particular $\iint_R \left(\left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right) dR$*, which is equivalent to saying that the first derivatives are square integrable. If the function* v *satisfies these conditions, then it is said to be* ***admissible****.*

Before we proceed further we shall introduce the **Hilbert space**, which is defined as follows.

Definition 11.2. A Hilbert space $\mathcal{H}$ is a real or complex inner product space that is also a complete metric space with respect to the distance function induced by the inner product.

This implies that $\mathcal{H}$, is a complex vector space on which there is an inner product $\langle x, y \rangle$ associating a complex number to each pair of elements x, y of $\mathcal{H}$ that satisfies the following properties:

(i) $\langle y, x \rangle = \overline{\langle x, y \rangle}$.
(ii) $\forall a, b \in \mathbb{C}$ then $\langle ax_1 + bx_2, y \rangle = a\langle x_1, y \rangle + b\langle x_2, y \rangle$.
(iii) $\langle x, x \rangle \geq 0$.

Given the definition of the Hilbert space, we shall use the following shorthand, $v \in \mathcal{H}_e^1$, where the subscript refers to satisfying the essential boundary condition and the superscript 1 refers to the property that the first derivatives is square integrable.

A couple of important features to note here are: (1) that we shall integrate piecewise and as such we do not have any worries about the edges on the polynomial surface; and (2) if $\alpha = \beta = 0$, then this implies that there is no S_2 term in (11.2).

Given all these notes, we shall now prove Theorem 11.1.

Proof. The proof of Theorem 11.1 depends on the ability to rewrite the Laplacian operator in terms of other vector derivative operators. We start with

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} \equiv -\nabla^2 u = -\nabla \cdot (\nabla u) = -\begin{pmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{pmatrix} \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{pmatrix},$$

where $\nabla\cdot$ is the **divergence** operator. We also have the identity

$$\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 \equiv \nabla u \cdot \nabla u = |\nabla u|^2.$$

The final identity we have is

$$\nabla \cdot (w \nabla u) = w \nabla^2 u + \nabla w \cdot \nabla u, \tag{11.3}$$

where w is some other scalar function.

Now we apply calculus of variation techniques that we presented in Chapter 5, where we introduce an arbitrary variation w and the parameter ε, which is sometimes referred to as a **label**, so that we can write $v = u + \varepsilon w$. We now consider

$$\nabla (u + \varepsilon w) \cdot \nabla (u + \varepsilon w) = \nabla u \cdot \nabla u + 2\varepsilon \nabla u \cdot \nabla w + \varepsilon^2 \nabla w \cdot \nabla w.$$

Hence

$$I[u + \varepsilon w] = I[u] + 2\varepsilon \left(\iint_R (\nabla u \cdot \nabla w + quw)\, dR - \iint_R f w dR + \int_{S_2} \alpha u w dS - \int_{S_2} \beta w dS \right) + \mathcal{O}\left(\varepsilon^2\right). \tag{11.4}$$

We now apply (11.3) to (11.4) to obtain

$$\iint_R \left(\nabla \cdot (w \nabla u) - w \nabla^2 u + quw - fw \right) dR + \int_{S_2} (\alpha u - \beta)\, w dS. \tag{11.5}$$

Before we can progress any further, we need to introduce the Gauss divergence theorem which describes the flow out minus the flow in. This is illustrated in the diagram in Fig. 11.2, where we consider a two-dimensional flow and have

$$\frac{\partial q_1}{\partial x} \delta x \delta y + \frac{\partial q_2}{\partial y} \delta x \delta y + \mathcal{O}\left(q_1^2, q_2^2\right) = Q \delta x \delta y, \tag{11.6}$$

where Q is the source strength. Next we introduce the vector $\tilde{\mathbf{q}} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$, divide by $\delta x \delta y$, and then let $\delta x, \delta y \to 0$, which results in

$$\nabla \cdot \tilde{\mathbf{q}} = Q. \tag{11.7}$$

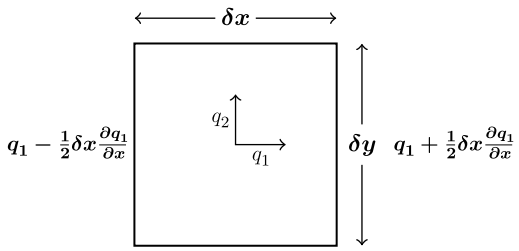


FIGURE 11.2

Example of the flow in-flow out domain.

The expression in (11.7) is referred to as the **continuity equation**. Next we integrate both sides of the equation in (11.7) over the interior of the domain, which gives

$$\iint_R \nabla \cdot \tilde{\mathbf{q}} dR = \iint_R Q dR = \text{ Total water/Heat coming into } R = \text{ Flow across surface of } R,$$

which implies

$$\iint_R \nabla \cdot \tilde{\mathbf{q}} dR = \int_S \tilde{\mathbf{q}} \cdot \mathbf{n} dS. \tag{11.8}$$

Returning to the functional in Theorem 11.1, we have $\tilde{\mathbf{q}} = w\nabla u$, $w\nabla u \cdot \mathbf{n} = w\frac{\partial u}{\partial \mathbf{n}}$, and $\nabla \cdot \tilde{\mathbf{q}} = \nabla \cdot (w\nabla u)$. Therefore, the coefficients of the 2ε term are

$$\int_{S_1+S_2} w\frac{\partial u}{\partial \mathbf{n}} dS + \iint_R w\left(-\nabla^2 u + qu - f\right) dR + \int_{S_2} w\left(\alpha u - \beta\right) dS,$$

which is equal to

$$\int_{S_1} w\frac{\partial u}{\partial \mathbf{n}} dS + \iint_R w\left(-\nabla^2 u + qu - f\right) dR + \int_{S_2} w\left(\frac{\partial u}{\partial \mathbf{n}} + \alpha u - \beta\right) dS. \tag{11.9}$$

If we have that u is the solution to problem A, then $-\nabla^2 u + qu - f \equiv 0$ in the domain R. We also have the mixed boundary conditions of the S_2 boundary, $\frac{\partial u}{\partial \mathbf{n}} + \alpha u - \beta = 0$, where we have $u = g$ on the S_1 boundary. Now if we consider the admissible variations of the form $u + \varepsilon w$, then this must satisfy the boundary condition on S_1, which implies that $u + \varepsilon w = g$, and can only happen if $w = 0$ on S_1. Therefore, given the reasoning above, we have that all of the coefficients of ε^2 are equal to zero.

We now consider the proof the other way where we now assume that if all of the coefficients of ε^2 are equal to zero then all of these terms must separately vanish because w is an arbitrary variation such that $u + \varepsilon w$ is admissible hence u is a solution. Also, the coefficient of ε^2,

$$\iint_R \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + qw^2 dR + \int_{S_2} \alpha w^2 dS \geq 0, \quad q, \alpha \geq 0,$$

is only zero if w is identically zero: the only constant value that w can have because $w = 0$ on S_1. Therefore, we have

$$I[u + \varepsilon w] = I[u] + \varepsilon^2 \left[\text{term which is greater than zero as long as } w \neq 0\right],$$
$$\text{thus } I[u + \varepsilon w] \geq I[u] \Rightarrow I[u + \varepsilon w] - I[u] \geq 0.$$

Therefore, $I[v]$ is minimized over all $v \in \mathcal{H}_e^1$ by $v = u$, which is the solution to problem A and thus we have proven Theorem 11.1.

Given all the information above, we know that the exact solution of problem A also minimizes the functional $I[v]$ over all $v \in \mathcal{H}_e^1(R)$. In finite elements we have an approximate solution u^h, where h relates to the size of the elements, which we obtain by minimizing $I[v]$ over S^h which is a **finite dimensional subspace** of $\mathcal{H}_e^1(R)$. The starting point is to take a trial solution, $v^h \in S^h$, which is defined as

$$v^h = \sum_{j=1}^{N} v_j^h \phi_j(x, y), \tag{11.10}$$

where $\phi_j(x, y)$ are **basis functions**, which span S^h, and v_j^h are parameters.

The basis functions in the Rayleigh-Ritz methods from the 1890s are global, which is equivalent to the Fourier series, whereas in FEM the basis functions are local. That is to say we take the domain R and divide into triangular elements whose vertices are the nodes, and the basis functions $\phi_j(x, y)$, are such that the approximation is **piecewise linear**, and the v_j^h are the model values of the approximation v^h.

We define the basis functions as

$$\phi_j(x, y) = \begin{cases} 1 & \text{at node } j, \\ 0 & \text{at other nodes,} \end{cases}$$

and we also have that $\phi_j(x, y) \equiv 0$ outside the neighborhood of node j. Given these conditions, we focus on a node, say j, to obtain ϕ_j; we know that we must have $\phi_j = 1$ at node j and then to be decreasing to zero toward the neighboring nodes; finally the basis function is identically zero; therefore, the shape is a tetrahedron. To help with description we have drawn a pyramid in Fig. 11.3A, where the pyramid is on the base formed between the nodes $lkmn$ which are the neighborhood of the node j, which is equal to the union of the elements where point j is a node.

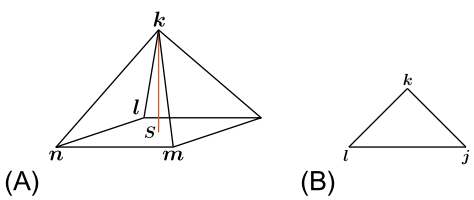


FIGURE 11.3

(A) a tetrahedron (pyramid) basis function, and (B) the specific triangle element between nodes j and l.

If we focus on the element with the triangle between nodes ljk, which is an equilateral triangle, as shown in Fig. 11.3B, then our numerical approximation is

$$v^h = v_j^h \phi_j + v_l^h \phi_l + v_k^h \phi_k, \tag{11.11}$$

where each basis function, $\phi_{(j,l,k)}$, is linear so this is a plane with height v_j^h, v_k^h, v_l^h at nodes j, k, l, respectively. There are no other basis functions that can affect this triangle so the net effect is piecewise linear and we have a polyhedral surface of flat plates. We should note here that the sum of the basis functions $\sum_j \phi_j(x, y) = 1$, because if $v_j^h = 1$ for all j then we would have a plane of height 1.

We now move on to how we obtain an approximation of problem A by minimizing $I\left[v^h\right]$ with respect to the parameter v_j^h. We start by assuming $\alpha = \beta = 0$, and $\frac{\partial v}{\partial \mathbf{n}} = 0$ on S_2, which leads to

$$I\left[v^h\right] = \iint_R \left(\nabla \left(\sum_j v_j^h \phi_j \right) \cdot \nabla \left(\sum_j v_j^h \phi_j \right) + q \sum_j \left(v_j^h \phi_j \right)^2 - 2f \sum_j v_j^h \phi_j \right) dR, \tag{11.12}$$

where there is no S_2 integral due to no flow; q is a constant. Therefore, we can take the parameters v_j^h and v_i^h out of the integrals, which enables (11.12) to be written as

$$\begin{aligned} I\left[v^h\right] &= \sum_{i=1}^{N}\sum_{j=1}^{N} v_j^h v_i^h \iint_R \nabla\phi_i \cdot \nabla\phi_j dxdy + q \sum_{i=1}^{N}\sum_{j=1}^{N} v_j^h v_i^h \iint_R \phi_i \phi_j dxdy - 2\sum_{j=1}^{N} v_j^h \iint_R f\phi_j dxdy, \\ &= \sum_{i=1}^{N}\sum_{j=1}^{N} k_{ij} v_i^h v_j^h + q \sum_{i=1}^{N}\sum_{j=1}^{N} m_{ij} v_i^h v_j^h - 2\sum_{j=1}^{N} v_j^h f_j, \end{aligned} \tag{11.13}$$

where

$$k_{ij} \equiv \iint_R \nabla\phi_i \cdot \nabla\phi_j dxdy, \qquad m_{ij} \equiv \iint_R \nabla\phi_i \cdot \nabla\phi_j dxdy, \qquad f_j = \iint_R f dxdy. \tag{11.14}$$

The elements presented in (11.14) define the following matrices: $\mathbf{K} = \left\{k_{ij}\right\}$ is the **stiffness matrix**, $\mathbf{M} = \left\{m_{ij}\right\}$ is the **mass matrix**, and $\boldsymbol{f} = \left\{f_j\right\}$ is the **load or external force vector**. The names of these matrices come from the mass on a spring problem where the tension $T = kx = mg$.

Given the definitions in (11.14), we can write (11.13) in terms of matrix-vector products as

$$I\left[\boldsymbol{v}^h\right] = \boldsymbol{v}^{h^T} \mathbf{K} \boldsymbol{v}^h + q \boldsymbol{v}^{h^T} \mathbf{M} \boldsymbol{v}^h - 2\boldsymbol{v}^{h^T} \boldsymbol{f}, \tag{11.15}$$

where $\boldsymbol{v}^h$ is the vector of the modal values $\boldsymbol{v}^h = \left\{v_j^h\right\}$.

An important feature to note here is that since

$$\boldsymbol{v}^{h^T} \mathbf{M} \boldsymbol{v}^h = \iint_R \left(\sum_{j=1}^{N} v_j \phi_j \right)^2 dxdy > 0, \quad \forall \boldsymbol{v}^h \neq \mathbf{0},$$

which is a positive definite quadratic form, as such this makes the mass matrix $\mathbf{M}$, **positive definite**, and so implies that all of its eigenvalues are greater than zero. We also have that

$$\boldsymbol{v}^{h^T}\mathbf{K}\boldsymbol{v}^h = \iint_R \left|\nabla \boldsymbol{v}^h\right|^2 dxdy \geq 0,$$

which implies that the stiffness matrix $\mathbf{K}$ is positive **semi-definite**. This is to say that $\mathbf{K}$ is singular when first assembled. However, we must recall that we have not used the *essential* boundary condition yet. This leads to the condition that we have to minimize $I\left[\boldsymbol{v}^h\right]$ with respect to v_j^h, which in turn leads to $\frac{\partial I\left[\boldsymbol{v}^h\right]}{\partial v_j^h} = 0$ for $j = 1, 2, \ldots, N$.

As an aside here, as a result of $\mathbf{M}$ being positive definite, and $\mathbf{K}$ being positive semi-definite, and if we have that $q \neq 0$, then the condition above is sufficient for a minimum.

To help illustrate this theory, we consider a general $N = 2$ situation where the functional is given by

$$I\left[\boldsymbol{v}^h\right] = k_{11}v_1^{h^2} + 2k_{12}v_1^h v_2^h + k_{22}v_2^{h^2} + q\left(m_{11}v_1^{h^2} + 2m_{12}v_1^h v_2^h + m_{22}v_2^{h^2}\right) - 2\left(f_1 v_1^h + f_2 v_2^h\right). \tag{11.16}$$

Differentiating (11.16) with respect to v_1^h yields

$$\frac{\partial I\left[\boldsymbol{v}^h\right]}{\partial v_1^h} = 2\left(k_{11}v_1^h + k_{12}v_2^h\right) + 2q\left(m_{11}v_1^h + m_{12}v_2^h\right) - 2f_1 = 0. \tag{11.17}$$

We now introduce the finite element solution of the form $\boldsymbol{u}^h \equiv \sum u_j^h \phi_j(x, y)$, which leads to

$$k_{11}u_1^h + k_{12}u_2^h + qm_{11}u_1^h + qm_{12}u_2^h = f_1, \tag{11.18a}$$

$$k_{12}u_1^h + k_{22}u_2^h + qm_{12}u_1^h + qm_{22}u_2^h = f_2, \tag{11.18b}$$

and this in turn leads to the general matrix-vector equation,

$$(\mathbf{K} + q\mathbf{M})\,\boldsymbol{u}^h = \boldsymbol{f}, \tag{11.19}$$

where $\boldsymbol{u}^h = \left\{u_j^h\right\}$ is a set of linear simultaneous equations for the u_j^h that are the values of u at the nodes and the finite element solution, $u^k(x, y) = \sum_{j=1}^N u_j^h \phi_j(x, y)$, which is a piecewise polyhedroid surface. This solution must satisfy the essential boundary condition for any node $j \in S_i$, and u_j^h is given.

When we consider the essential boundary conditions, for the natural boundary conditions $\frac{\partial u}{\partial \mathbf{n}} = 0$ on S_2 we do not have to consider any changes. The Neumann boundary conditions on S_1 can be useful and we shall see how later. We now consider the mixed boundary conditions where we have $\frac{\partial u}{\partial \mathbf{n}} + \alpha u = \beta$ on S_2 as follows. When we have the case of $\alpha \neq 0$, this introduces an extra term in $I\left[v^h\right] = \int_{S_2} \alpha\left(\sum_j v_j^h \phi_j\right)^2 dS$. Differentiating with respect to v_j^h v_j^h implies $\int_{S_2} \alpha\phi_j \sum_j v_j^h \phi_j dS$, and is only non-zero for $l \in S_2$ and nodes j in the neighborhood of node l on S_2, as we show in Fig. 11.4, where we see that the integral is along S_2.

Therefore, we have extra terms in the lth row of $\mathbf{K} + q\mathbf{M}$ which is on the left-hand side of the simultaneous linear equations.

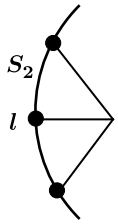


FIGURE 11.4

Boundary integral illustration.

If we now consider the case where the mixed boundary conditions have $\beta \neq 0$, then we have an extra linear term in $I\left[v^h\right]$ as $-2\int_{S_2}\beta\sum_j v_j^h\phi_j dS$. Next, differentiating with respect to v_j^h results in $\int_{S_2}\beta\phi_j dS$, which creates an extra term on the right-hand side of the simultaneous equations.

Remark 11.3. The solution of problem A minimizes the functional $I[v]$, but there are many problems that do not have a functional form; one such partial differential equation is $\frac{\partial u}{\partial t} = \sigma\frac{\partial^2 u}{\partial t}$ which is the parabolic heat conduction or groundwater flow in one dimension.

Recall that we have $(\mathbf{K}+q\mathbf{M})\boldsymbol{u}^h = \boldsymbol{f}$ from minimizing the functional $I\left[\boldsymbol{v}^h\right]$ for the approximate solution of problem A; we can then obtain the same result by taking a **weak form** of a differential equation where this technique is always possible, even if there is not a functional form.

11.2 Weak Solutions of Differential Equation

If we consider the second-order partial differential equation

$$-\nabla^2 u + qu = f, \tag{11.20}$$

then the **exact** solution to (11.20) is referred to as the **classical** solution.

We start by multiplying (11.20) by a *test function*, $w(x, y)$, and integrate over the domain R, which results in

$$\iint_R w\left(-\nabla^2 u + qu\right)dR = \iint_R f w dR. \tag{11.21}$$

This is one weak form. We could have considered using an approximation for u, but this implies at least piecewise quadratic. If we now integrate (11.21) by parts, using (11.3), then this results in

$$\begin{aligned}\iint_R (\nabla w\cdot\nabla u + quw)\,dR - \iint_R \nabla\cdot(w\nabla u)\,dR &= \iint_R f w dR,\\ \iint_R (\nabla w\cdot\nabla u + quw)\,dR - \int_{S_1+S_2} w\frac{\partial u}{\partial \mathbf{n}}dS &= \iint_R f w dR,\end{aligned} \tag{11.22}$$

and we should note u is still a solution of problem A. We now use mixed boundary conditions on the S_2 boundary, which results in

$$\iint_R (\nabla w \cdot \nabla u + quw)\, dR - \int_{S_1} w \frac{\partial u}{\partial \mathbf{n}} dS + \int_{S_2} \alpha uw dS = \iint_R f w dR + \int_{S_2} \beta w dS. \tag{11.23}$$

We have another weak form with only the first derivatives present, hence we approximate (11.23) with a piecewise linear solution of the form $v^h = \sum_{i=1}^{N} v_i^h \phi_i(x, y)$ and take $w = \phi_j(x, y)$ for $j = 1, 2, \ldots, N$. The result of this is a **weak solution**. A feature to note here is that this solution is equivalent to an **integral transform**: for example, those involved in **Fourier and Laplace transforms**. We shall go into more detail about these two specific transforms in Chapter 12.

We now have a matrix-vector equation of the form

$$(\mathbf{K} + q\mathbf{M})\boldsymbol{u}^h + \boldsymbol{g} = \boldsymbol{f}, \tag{11.24}$$

where the $\mathbf{K}$ matrix contains the α terms, and $\boldsymbol{g} \equiv \int_{S_1} \phi_j \frac{\partial u}{\partial \mathbf{n}} dS$. For the u_j^h, where $j \in S_1$ are known through the boundary conditions, then the equations for these points are removed from the matrix-vector equation and their specific values are substituted back into the remaining equations that contain them. After the rest of the equations are solved for the unknown u_j^h, these values are substituted back to obtain an approximation to the terms in (11.24).

The approach just described is useful in problems where we want to estimate the balance of flow. If we take $w = \phi(x, y)$, then the basis functions can be referred to by any of the following four names:

- standard Galerkin;
- Rayleigh-Ritz;
- virtual work (stress analysis); or
- Baublov Galerkin.

If, however, we have $v^h = \sum_i v_i^h \phi_i(x, y)$ and $w \neq \phi_j$, then this technique is referred to as the **Petrov-Galerkin** approach. As an aside, these functions are often referred to as *weights* in the engineering literature.

We now introduce an example, as shown in Fig. 11.5 to help illustrate the all of the theory presented so far.

We are solving the following differential equation:

$$-\frac{d^2u}{dx^2} + qu = a \sin \frac{\pi x}{2}, \tag{11.25}$$

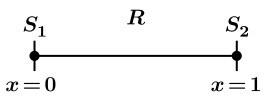


FIGURE 11.5

Domain for the first finite element example.

for $(0 < x < 1)$ with $u(0) = b$ at $S_1 \equiv x = 0$ and mixed boundary conditions $\frac{\partial u}{\partial x} + \alpha u = \beta$ at $S_2 \equiv x = 1$. The $\frac{du}{dx}$ term at $x = 1$ is the outward normal derivative; we also have that q is equal to a constant that is greater than or equal to zero, and α is greater than or equal to zero.

There are two methods that we can use to solve this differential equation. Here we consider the following approach of minimizing the functional:

$$I[v] = \int_0^1 \left[\left(\frac{dv}{dx} \right)^2 + qv^2 - 2fv \right] dx + \left[\alpha v^2 - 2\beta v \right]_{x=1},$$

over $v \in \mathcal{H}_e^1$, $v = b$ at $x = 0$ is the essential boundary condition, where the second approach is to solve the weak form through multiplying (11.25) by $w(x)$ and integrating by parts in one dimension, which yields

$$\int_0^1 w \left(-\frac{d^2u}{dx^2} + qu \right) dx = \int_0^1 fw dx, \tag{11.26}$$

$$\int_0^1 \left[-\frac{d}{dx}\left(w\frac{du}{dx} \right) + \frac{dw}{dx}\frac{du}{dx} + quw \right] dx = \int_0^1 fw dx,$$

$$-\left[w\frac{du}{dx} \right]_{x=1} + \left[w\frac{du}{dx} \right]_{x=0} + \int_0^1 \frac{dw}{dx}\frac{du}{dx} + quw dx = \int_0^1 fw dx,$$

$$-\left[x(-\alpha + \beta) \right] + w\left[\frac{du}{dx} \right]_{x=0} + \int_0^1 \frac{dw}{dx}\frac{du}{dx} + quw dx = \int_0^1 fw dx, \tag{11.27}$$

is a weak form.

We now divide the line $[0, 1]$ into two unequal linear elements with lengths h_1 and h_2, respectively. Next we form the solution

$$u \approx \sum_{j=0}^{2} u_j \phi_j(x) \equiv u_0\phi_0(x) + u_1\phi_1(x) + u_2\phi_2(x),$$

where we have the following rules:

$\phi_j = 1$ at node j;

$\phi_j = 0$ at all neighborhood of j; and

$\phi_j \equiv 0$ outside the neighborhood of node j.

We have drawn a diagram of the conditions above for the problem that we are consider here in Fig. 11.6. Therefore the basis functions for this problem that satisfy the conditions above are

$$\phi_0 = 1 - \frac{x}{h_1}, \qquad 0 \le x \le h_1, \tag{11.28a}$$

$$\phi_1 = \begin{cases} \frac{x}{h_1}, & 0 \le x \le h_1, \\ \frac{1-x}{h_2}, & h_1 \le x \le 1, \end{cases} \tag{11.28b}$$

$$\phi_2 = \frac{x - h_1}{h_2}, \qquad h_1 \le x \le 1. \tag{11.28c}$$

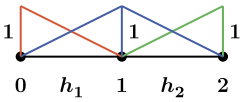


FIGURE 11.6

Example of linear basis functions.

If we now apply a standard Galerkin method, then we put $w = \phi_k(x)$, $k = 0, 1, 2$ in turn. We need to note that the α and β terms only apply for $x = 1$, which implies that they are only considered for $k = 2$.

For $k = 0$, then $w = \phi_0(x)$, such that

$$\int_0^{h_1} \left(u_0\phi_0' + u_1\phi_1' + u_2\phi_2'\right)\phi_0' dx + q\int_0^{h_1} \left(u_0\phi_0 + u_1\phi_1\right)\phi_0 dx + \left[\phi_0 \frac{du}{dx}\right]_{x=0} = \int_0^{h_1} f\phi_0 dx, \tag{11.29}$$

where we should note that the neighborhood of node 0 is e_1 from the diagram in Fig. 11.6 and as such $\phi_2 \equiv 0$ in the neighborhood of node $j = 0$.

Exercise 11.4. *Verify that the integral of* (11.29) *with the definition of the basis functions for the first segment is*

$$\frac{1}{h_1}(u_0 - u_1) + \frac{q_1 h_1}{6}(2u_0 + u_1) + \left[\frac{du}{dx}\right]_{x=0} = \int_0^{h_1} f\phi_0 dx. \tag{11.30}$$

If we now consider the second node, $k = 1$, then we have

$$\int_0^{h_1} \left(u_0\phi_0' + u_1\phi_1'\right)\phi_1' dx + \int_{h_1}^{1} \left(u_1\phi_1' + u_2\phi_2'\right)\phi_1' dx + q\int_0^{h_1} \left(u_0\phi_0 + u_1\phi_1\right)\phi_1 dx$$
$$+ \int_{h_1}^{1} \left(u_1\phi_1 + u_2\phi_2\right)\phi_1 dx = \int_0^{1} f\phi_1 dx, \tag{11.31}$$

where the neighborhood of the node $j = 1$ is $e_1 \cup e_2$.

Exercise 11.5. *Verify that the integral in* (11.31) *integrates to*

$$\frac{1}{h_1}(-u_0 + u_1) + \frac{1}{h_2}(u_1 - u_2) + \frac{qh_1}{6}(u_0 - 2u_1) + \frac{qh_2}{6}(2u_1 + u_2) = \int_0^1 f\phi_1 dx. \tag{11.32}$$

Hint: Remember that $h_1 + h_2 = 1$.

We finally consider the third node, $j = 2$, where we now have to solve

$$\int_{h_1}^{1} \left(u_1\phi_1' + u_2\phi_2'\right)\phi_2' dx + q\int_{h_1}^{1} \left(u_1\phi_1 + u_2\phi_2\right)\phi_2 dx - \left[\phi_2 \frac{du}{dx}\right]_{x=1} = \int_{h_1}^{1} f\phi_2 dx, \tag{11.33}$$

where the neighborhood for this node is e_2.

Exercise 11.6. *Verify that the integral in* (11.33) *integrates to*

$$\frac{1}{h_2}(-u_1 + u_2) + \frac{qh_2}{6}(u_1 + 2u_2) + \alpha u_2 = \int_{h_1}^{1} f\phi_2 dx + \beta. \tag{11.34}$$

Remember to use the mixed boundary conditions at $x = 1$.

Given the expressions in (11.30), (11.32), and (11.34), we can now assemble the matrix-vector equation, which can easily be shown to initially be

$$\begin{pmatrix} \frac{1}{h_1} & -\frac{1}{h_1} & 0 \\ -\frac{1}{h_1} & \frac{1}{h_1}+\frac{1}{h_2} & -\frac{1}{h_2} \\ 0 & -\frac{1}{h_2} & \frac{1}{h_2}+\alpha \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} + \frac{q}{6} \begin{pmatrix} 2h_1 & h_1 & 0 \\ h_1 & 2h_1+2h_2 & h_2 \\ 0 & h_2 & 2h_2 \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} + \begin{pmatrix} \left(\frac{du}{dx}\right)_{x=0} \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} F_0 \\ F_1 \\ F_2 \end{pmatrix}, \tag{11.35}$$

where $F_j \equiv \int_0^1 \phi_j f\, dx$ for $j = 0, 1, 2$, and the first matrix in (11.35) is the stiffness matrix for this problem and the second matrix is the mass matrix. Note that the α term in the stiffness matrix arises due to the fact that node $j = 2$ is on the boundary S_2. We should also note that if we have the case where α was equal to zero, then the stiffness for this problem would be singular.

If we consider the Neumann problem associated with this example, then we would be solving the case where $q = 0$ and $f = 0$, which would leave $\frac{d^2u}{dx^2} = 0$ with $\frac{du}{dx} = 0$ at $x = 0$ and $x = 1$. The classical solution in its general form for this problem is $u = Cx + d$, but $C = 0$ which leaves $u = d$, where d is arbitrary, and therefore there is no unique classical solution for this problem. The corresponding finite element equations for this situation would just be the stiffness matrix multiplying the vector $\boldsymbol{u}$, but the stiffness matrix would be singular.

We now consider the two-dimensional case where we have

$$-\nabla^2 u = 0 \text{ in } R,$$
$$\frac{\partial u}{\partial \mathbf{n}} = 0 \text{ on boundary of } R.$$

The solution to the partial differential equation above is u equal to any constant, where this type of partial differential equation occurs in the floating membrane problem. In order to define a unique solution, we require at least one S_1 point. If we had the situation where $u_0 = b$ then the first equation is removed from the matrix-vector formulation, where we now substitute $u_0 = b$ into the remaining equations that contain u_0.

Returning to our example, we still have the problem that we do not know $\left(\frac{du}{dx}\right)_{x=0}$ due to the fact that the node $j = 0$ is on the S_1 boundary, so we shall remove the equation associated with $j = 0$, and substitute $u_0 = b$ into the remaining equations, which yields

$$\begin{pmatrix} \frac{1}{h_1}+\frac{1}{h_2} & -\frac{1}{h_2} \\ -\frac{1}{h_2} & \frac{1}{h_2}+\alpha \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \frac{q}{6} \begin{pmatrix} 2(h_1+h_2) & h_2 \\ h_2 & 2h_2 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} F_1 + \left(\frac{1}{h_1} - \frac{qh_1}{6}\right) b \\ F_2 + \beta \end{pmatrix}. \tag{11.36}$$

We therefore have two simultaneous equations for the two unknowns, u_1 and u_2; where upon solving for these, we substitute these values into the equation for $j = 0$ to obtain $\left(\frac{du}{dx}\right)_{x=0}$.

It is also possible to obtain an estimate of the gradient at an interior node by the following device. We integrate over **one element only** on either side of this node. To achieve this goal, we return to the weak form, but we shall integrate over e_2 only. This leads to

$$\int_{h_1}^{h_1+h_2} w\left(-\frac{d^2u}{dx^2} + qu\right) dx = \int_{h_1}^{h_1+h_2} f w\, dx,$$

$$\int_{h_1}^{h_1+h_2}\left(\frac{d}{dx}\left(w\frac{du}{dx}\right)+\frac{dw}{dx}\frac{du}{dx}+quw\right)dx=\int_{h_1}^{h_1+h_2}fwdx,$$
$$-\left[w\frac{du}{dx}\right]_{x=h_1+h_2}+\left[w\frac{du}{dx}\right]_{x=h_1}+\int_{h_1}^{h_1+h_2}\left(\frac{dw}{dx}\frac{du}{dx}+quw\right)dx=\int_{h_1}^{h_1+h_2}fwdx. \tag{11.37}$$

As before we set $u=\sum_{j=1}^{2}u_j\phi_j$ and $w=\phi_1$; we have plotted a diagram of this situation to illustrate the shape of the element that we are considering here in Fig. 11.7. Recalling the definition of ϕ_1 in e_2, we have $\phi_1=\frac{1-x}{h_2}\equiv\frac{h_1+h_2-x}{h_2}$, where $\phi_1=0$ at $x=h_1+h_2$ and $\phi_1=1$ at $x=h_1$. With this information, we obtain

$$\left[\frac{du}{dx}\right]_{x=h_1}+\frac{1}{h_2}(u_1-u_2)+\frac{qh_2}{6}(2u_1+u_2)=\int_{h_1}^{h_1+h_2}f\phi_1dx. \tag{11.38}$$

If we substituted our values for u_1 and u_2 into (11.38), then we have an estimate for $\left[\frac{du}{dx}\right]_{x=h_1}$.

11.2.1 Heat Development Due to Hydration of Concrete

In the situation that we consider here, we have the parabolic partial differential equation

$$c\frac{\partial u}{\partial t}=\frac{\partial^2u}{\partial x^2}+Qe^{-\alpha t}\text{ for }0<x\leq 1, \tag{11.39}$$

where we have a large slab with unit thickness and $u=u(x,t)$ is the temperature; we also have that $t>0$ and Q, α, and c are all positive constants. The boundary constants are $u(0,t)=u(1,t)=0$, where the outside temperature is taken as a datum and we have the initial conditions $u(x,0)=0$.

To obtain the solution to (11.39), we shall multiply (11.39) by $w(x)$ and integrate by parts in space, which results in

$$c\int_0^1 w\frac{\partial u}{\partial t}dx=\int_0^1 w\frac{\partial^2u}{\partial x^2}dx+\int_0^1 wQe^{-\alpha t}dx,$$
$$=\left[w\frac{\partial u}{\partial x}\right]_{x=1}-\left[w\frac{\partial u}{\partial x}\right]_{x=0}-\int_0^1\frac{dw}{dx}\frac{\partial u}{\partial x}dx+qe^{-\alpha t}\int_0^1 wdx. \tag{11.40}$$

We now introduce two linear equal elements where we have $h_1=h_2=\frac{1}{2}$ and put $u\approx\sum_{j=0}^{2}u_j(t)\phi_j(x)$, which is equivalent to the separation of variables. We next notice that we have $\phi_j(x)$ that are

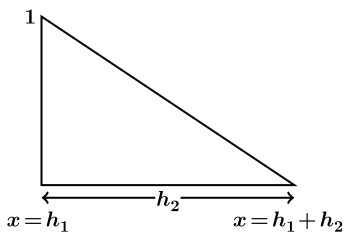


FIGURE 11.7

Schematic of the interior gradient function.

piecewise linear as in the previous example, where the modal values are $u_j(t)$. We shall set $w = \phi_k(x)$ in turn for $k = 0, 1, 2$, which results in

$$c\sum_{j=0}^{2}\frac{du_j}{dt}\int_0^1 \phi_k(x)\,\phi_j(x)\,dx + \sum_{j=0}^{2} u_j(t)\int_0^1 \frac{d\phi_k}{dx}\frac{d\phi_j}{dx}dx - \left[\phi_k\frac{\partial u}{\partial x}\right]_{x=1} + \left[\phi_k\frac{\partial u}{\partial x}\right]_{x=0} = Qe^{-\alpha t}\int_0^1 \phi_k dx, \quad k = 0, 1, 2. \tag{11.41}$$

We now assemble the stiffness and mass matrices for (11.41), which results in

$$c\mathbf{M}\frac{d}{dt}\boldsymbol{u}^h + \mathbf{K}\boldsymbol{u}^h + \boldsymbol{g} = \boldsymbol{f}, \tag{11.42}$$

where $\boldsymbol{g}$ is a vector containing the boundary gradient terms, and upon applying the integrals in (11.41) we obtain

$$\frac{c}{6}\begin{pmatrix} 2h & h & 0 \\ h & 4h & h \\ 0 & h & 2h \end{pmatrix}\begin{pmatrix} \dot{u}_1 \\ \dot{u}_2 \\ \dot{u}_3 \end{pmatrix} + \begin{pmatrix} \frac{1}{h} & -\frac{1}{h} & 0 \\ -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} \\ 0 & -\frac{1}{h} & \frac{1}{h} \end{pmatrix}\begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} + \begin{pmatrix} \left[\frac{\partial u}{\partial x}\right]_{x=0} \\ 0 \\ -\left[\frac{\partial u}{\partial x}\right]_{x=1} \end{pmatrix} = Qe^{-\alpha t}\begin{pmatrix} \frac{h}{2} \\ h \\ \frac{h}{2} \end{pmatrix}. \tag{11.43}$$

The next step is to overwrite the first and the last rows in (11.43), because at the moment the stiffness matrix in (11.42) is singular. We have the condition that $u_0 = u_2 = 0$, along with $\dot{u}_0 = \dot{u}_2 = 0$, so upon substituting these values in the middle row in (11.43) results in the ordinary differential equation for u_1:

$$\frac{c}{3}\frac{du_1}{dt} + 4u_1 = \frac{1}{2}Qe^{-\alpha t}. \tag{11.44}$$

The classical solution to (11.44) is an infinite series which is obtained though the separation of variables technique.

11.2.2 Torsion of a Bar of Equilateral Triangle Cross Section

The diagram to illustrate this example is presented in Fig. 11.8, where we are solving the partial differential equation

$$-\nabla^2 u = 2 \text{ in } R,$$

where $u = u(x, y)$ is the stress function, and the gradients $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ give the stress in the material.

The domain R here is an equilateral triangle with length a, where we have $u = 0$ on the boundaries. As the domain is symmetric, we can solve for a solution on one-half of the triangle and then use triangle elements; a diagram for this formulation is presented in Fig. 11.9. We therefore wish to solve the functional

$$\iint_{\triangle} \nabla\phi_i \cdot \nabla\phi_j dxdy, \tag{11.45}$$

for a general linear triangle.

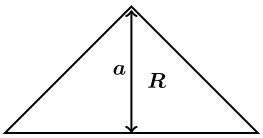


FIGURE 11.8

Triangular torsion problem domain.

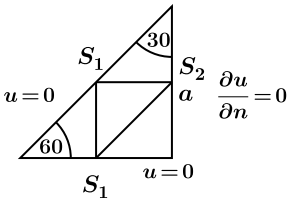


FIGURE 11.9

Triangle elements for the torsion problem domain.

For us to be able to solve this problem, we wish to use linear triangle elements. If we consider the two diagrams in Fig. 11.10, which show the general triangle element and then the same triangle but with a new set of axes applied with respect to ξ and η, then we require the basis function associated with point A, ϕ_A to be equal to one at this point and then equal to zero by the time we arrive at points B and C, but should be identically equal to zero on the line CB.

Given this setup, the question becomes: What is $|\nabla\phi_A|$? This gradient is invariant under a rotation of axes, which implies that

$$\left(\frac{\partial\phi}{\partial x}\right)^2+\left(\frac{\partial\phi}{\partial y}\right)^2=\left(\frac{\partial\phi}{\partial\xi}\right)^2+\left(\frac{\partial\phi}{\partial\eta}\right)^2.$$

However, $\frac{\partial\phi_A}{\partial\xi}=0$ which leaves $|\nabla\phi_A|=\left|\frac{\partial\phi}{\partial\eta}\right|=\frac{1}{b\sin c}$. Therefore we have

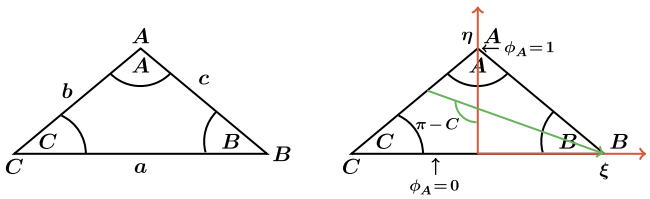


FIGURE 11.10

Illustration of the coordinate change from global to local for the triangular elements.

$$\iint \triangle |\nabla \phi_A|^2 \, dxdy = \frac{\triangle ABC}{b^2 \sin^2 C} = \frac{\frac{1}{2}ab \sin C}{b^2 \sin^2 C} = \frac{\frac{1}{2}b \cos C + c \cos B}{b \sin C},$$
$$= \frac{1}{2}(\cot C + \cot B), \tag{11.46}$$

where we have to calculate the area of the triangle ABC, which is half the base times the height, and note that $b \sin C = c \sin B$. We also have

$$\iint_{\triangle} \nabla \phi_A \cdot \nabla \phi_B dxdy \equiv \iint_{\triangle} |\nabla \phi_A| \, |\nabla \phi_B| \cos(\pi - C) \, dxdy = -\frac{\triangle ABC}{b \sin cC \sin A} \cos C,$$
$$= -\frac{1}{2}\frac{bc \sin A \cos C}{bc \sin A \sin C} = -\frac{1}{2}\cot C. \tag{11.47}$$

Inside the triangle, ABC, the finite element approximation is $u_A\phi_A + u_B\phi_B + u_C\phi_C$. The contribution from the area of the triangle to the global (assembled) stiffness matrix is then given by

$$\iint_{\triangle} (u_A\phi_A + u_B\phi_B + u_C\phi_C) \cdot \nabla \phi_k dxdy, \quad \text{for } k = A, B, C \text{ in turn.} \tag{11.48}$$

Thus the **element stiffness matrix** is given by

$$\mathbf{K}_{e_i} = \begin{pmatrix} \iint_{\triangle} \nabla\phi_A \cdot \nabla\phi_A dxdy & \iint_{\triangle} \nabla\phi_A \cdot \nabla\phi_B dxdy & \iint_{\triangle} \nabla\phi_A \cdot \nabla\phi_C dxdy \\ \iint_{\triangle} \nabla\phi_A \cdot \nabla\phi_B dxdy & \iint_{\triangle} \nabla\phi_B \cdot \nabla\phi_B dxdy & \iint_{\triangle} \nabla\phi_C \cdot \nabla\phi_B dxdy \\ \iint_{\triangle} \nabla\phi_A \cdot \nabla\phi_C dxdy & \iint_{\triangle} \nabla\phi_B \cdot \nabla\phi_C dxdy & \iint_{\triangle} \nabla\phi_C \cdot \nabla\phi_C dxdy \end{pmatrix},$$
$$= \frac{1}{2}\begin{pmatrix} \cot B + \cot C & -\cot C & -\cot B \\ -\cot C & \cot C + \cot A & -\cot A \\ -\cot B & -\cot A & \cot A + \cot B \end{pmatrix}. \tag{11.49}$$

We now move on to form the global stiffness matrix, which involves collecting all of the element stiffness matrices. We must be careful to consistently label the nodes A, B, and C for each element. The largest x value should determine the A node; however, if there should be two nodes at the same level, then we start with the node with the largest y value and move clockwise. We must also introduce **dual labeling** as nodes are shared with adjacent triangles. If we denote *node number* by NN, then we have NN (element, local node label) = global node number.

If we consider the triangle domain that has been cut in half due to the symmetry of the problem, then in Fig. 11.11 where we have drawn and labeled the four triangular elements that cover the domain along with the associated six nodes, we only have to consider solving on one-half of the domain.

If we consider the two left triangles given by e_1 and e_2 in Fig. 11.11, then we have the following labeling:

$$\begin{aligned} NN(1, A) &= 1, & NN(2, A) &= 2, \\ NN(1, B) &= 3, & NN(2, B) &= 5, \\ NN(1, C) &= 2, & NN(2, C) &= 4. \end{aligned}$$

To start to fill in the global stiffness matrix, we require the cotangent of the three angles of the triangle, which are given by

$$\cot(30) = \sqrt{3}, \qquad \cot(60) = \frac{1}{\sqrt{3}}, \qquad \cot(90) = 0.$$

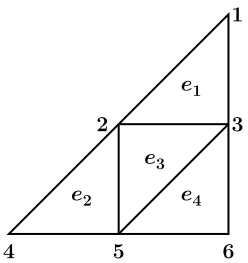


FIGURE 11.11

Finite element grid for the torsion problem.

We now label the entries in the element stiffness matrix as

$$\mathbf{K}_{e_i} \equiv \begin{pmatrix} K_{aa} & K_{ab} & K_{ac} \\ K_{ba} & K_{bb} & K_{bc} \\ K_{ca} & K_{cb} & K_{cc} \end{pmatrix},$$

where $K_{aa} = \frac{1}{2}(\cot B + \cot C)$ is an example of an entry in the element stiffness matrix.

We next form the global stiffness matrix for the torsion problem, but we must note that nodes 2 and 3 are part of three elements each and such they each contribute to that entry in the global stiffness matrix. Thus the generalized global stiffness matrix for a half triangular domain is

$$\mathbf{K}_g = \begin{pmatrix} K^1_{aa} & K^1_{ac} & K^1_{ab} & 0 & 0 & 0 \\ K^1_{ca} & K^1_{cc} + K^2_{aa} + K^3_{cc} & K^1_{cb} + K^3_{ca} & K^2_{ac} & K^2_{ab} + K^3_{cb} & 0 \\ K^1_{ba} & K^1_{bc} + K^3_{ac} & K^1_{bb} + K^3_{aa} + K^4_{aa} & 0 & K^3_{ab} + K^4_{ac} & K^4_{ab} \\ 0 & K^2_{ca} & 0 & K^2_{cc} & K^2_{cb} & 0 \\ 0 & K^2_{ba} + K^3_{bc} & K^3_{ba} + K^4_{ca} & K^2_{bc} & K^2_{bb} + K^3_{bb} + K^4_{cc} & K^4_{cb} \\ 0 & 0 & K^4_{ba} & 0 & K^4_{bc} & K^4_{bb} \end{pmatrix}, \tag{11.50}$$

where we have used the numbers in the superscript to represent the element e_i for $i = 1, 2, 3, 4$. Given the global stiffness matrix in (11.50), we would form the matrix-vector equation $\mathbf{K}_g \boldsymbol{u} + \boldsymbol{g} = \boldsymbol{f}$. However, you may have already spotted something: all but one of our nodes are not on a boundary. Therefore, we already know the values of u at these nodes. We also have that the boundary condition for node 3, g_3, is equal to zero, therefore we only have to evaluate

$$\frac{1}{2}(\cot 30 + \cot 60 + \cot 90 + \cot 30 + \cot 60 + \cot 90)\, u_3 = f_3.$$

We now have to evaluate f_3, which is given by

$$f_3 \equiv 2 \iint_{R_3} \phi_3 dx dy,$$

where R_3 represents the neighborhood of node 3, and the integral above is equivalent to two-thirds the height of the basis function, which is 1, times the area of the neighborhood of node 3, which is equivalent to the volume of a pyramid. The three triangles that are in the neighborhood of node 3 all have the same area, $\triangle \equiv \frac{a^2}{8\sqrt{3}}$, which comes from finding the base of the triangle through the trigonometric functions definitions. This makes $f_3 = \frac{a^2}{4\sqrt{3}}$, which leads to $\frac{4}{\sqrt{3}}u_3 = \frac{a^2}{4\sqrt{3}} \Rightarrow u_3 = \frac{a^2}{16}$.

11.3 Accuracy of the Finite Element Approach

In the theory presented so far, we have shown two different approaches to FEM:

1. functional approach, which is of the form $LU = f$; and
2. weak form—multiplying by $w(x)$ and integrating by parts.

This raises the question of why we should bother with the functional approach. We know that we have a functional to minimize when the differential operator L is symmetric and positive definite, as in $L \equiv -\nabla^2 + q$ for $q \geq 0$.

For example, suppose that we have homogeneous boundary conditions, that is to say, $u = 0$ on S_1 and $\frac{\partial u}{\partial \mathbf{n}} = 0$ on S_2. Now let

$$(Lu, v) \equiv \int_R v\,(Lu)\,dR,$$

which is for any dimensions, then we would have for the differential operator defined above

$$\begin{aligned}(Lu, v) &= \left(-\nabla^2 u + qu, v\right), \quad (q \geq 0), \\ &= (\nabla u, \nabla v) + (qu, v), \quad v = 0 \text{ on } S_1, \\ &= \left(u, -\nabla^2 v\right) + (u, qv), \\ &= (u, Lv),\end{aligned}$$

which implies that the differential operator L is symmetric, but also that the divergence and the grad operators $\nabla\cdot$ and ∇, respectively, are adjoint operators.

We also have the property that

$$(Lu, u) \equiv \int_R \left(|\nabla u|^2 + qu^2\right) dR > 0, \tag{11.51}$$

unless $u \equiv 0$, which implies that the differential operator is positive definite. This means that we have a natural way of measuring how good an approximation to the differential equation is. If we try to minimize the functional $I[v]$, then we can say that if $I[v_1] < I[v_2]$, then this implies that v_1 is a *better* approximation than v_2.

If we consider a linear piecewise approximation, then we would expect the accuracy of this approximation to have an error which is proportional to h^2 times some norm of a second derivative term. If

we consider the Taylor series of u, then we have

$$u(a+h) = u(a) + hu'(a) + \left| \frac{h^2}{2} u''(a+\theta h) \right|, \tag{11.52}$$

where the second term is the remainder and h is a measure of the size of the element.

We now suppose that the differential equation $Lu = -\nabla^2 u + gq = f$ as in problem A, but this time we have that $u = 0$ on the S_1 boundary. Note that it is always possible to introduce a change of function so that the 0 condition on the S_1 boundary can be met. Given this setup, we can define a functional for this problem as

$$I[v] = \underbrace{\iint_R \left(\left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 + qv^2 \right) dR + \int_{S_2} \alpha v^2 dS}_{a(v,v)} \underbrace{- 2 \left(\iint_R fv dR + \int_{S_2} \beta v dS \right)}_{-2(f,v)}, \tag{11.53}$$

where $a(v, v)$ is referred to as the internal quadratic term, and $-2(f, v)$ is a linear external term [417], where $v \in \mathcal{H}_e^1$, and the essential boundary condition is $v = 0$ on S_1. We now have a Hilbert space with the norm $v = \sqrt{a(v, v)} = \|v\|_{energy}$, with the induced inner product

$$a(v, w) = \iint_R (\nabla v \cdot \nabla w + qvw) dR + \int_{S_2} \alpha vw dS. \tag{11.54}$$

We should note here that the norm, $\|v\|_{en}$, is a natural norm which can include the variable's physical properties. Then if we have that u is the classical solution, and that u^h is the finite element solution and if we define the error as $e^h = u - u^h$, then the **energy in the error**, that is to say, $a\left(e^h, d^h\right)$, is a **minimum**. This can be interpreted as saying that the finite element solution minimizes the error as measured by the energy norm.

We now introduce the following theorem:

Theorem 11.7. *Suppose that u minimizes the function $I[v]$ over the Hilbert space $\mathcal{H}_e^1$ and S^h is the finite element solution, then:*

- **(a)** *The minimum of $I\left[v^h\right]$ and the minimum if $A\left(v - v^h, v - v^h\right)$ over $v^h \in S^h$ are given by $v^h = u^h$ which is the finite element solution.*
- **(b)** *With respect to the space with the inner product $a(v, w)$, u^h is the projection of u on S^h, that is, that $a\left(e^h, v^h\right) = 0$, $\forall v^h \in S^h$.*
- **(c)** *u^h satisfies $a\left(u^h, v^h\right) = \left(f, v^h\right)$ $\forall v^h \in S^h$ and if S^h tend to the whole space $\mathcal{H}_e^1$, then $a(u, v) = (f, v)$, $\forall v \in \mathcal{H}_e^1$ which is the **weak form**.*

Proof. We work in a space where the square of the length of v is equal to $a(v, v) = \|v\|_{en}^2$. Therefore, if we have $a(v, w) = 0$, then this implies that v and w are orthogonal. Since S^h is finite dimensional, it is **closed**, which implies that the limited number of points in S^h remain in S^h. Hence, there exists an $u^h \in S^h$, that minimizes $I[v]$ over $v \in S^h$, where

$$I\left[u^h\right] \leq I\left[u^h + \varepsilon v^h\right], \quad v^h \in S^h,$$

where u^h, v^h, and $u^h + \varepsilon v^h$ are all admissible. Therefore, using calculus of variation theory, we have

$$\begin{aligned} I\left[u^h\right] \leq I\left[u^h + \varepsilon v^h\right] &= I\left[u^h\right] + 2\varepsilon\left(a\left(u^h, v^h\right) - \left(f, v^h\right)\right) + \varepsilon^2 a\left(v^h, v^h\right), \\ 0 \leq 2\varepsilon\left(a\left(u^h, v^h\right) - \left(f, v^h\right)\right) &+ \varepsilon^2 a\left(v^h, v^h\right), \quad \forall\varepsilon, \end{aligned} \tag{11.55}$$

but the expression above is quadratic in ε with the coefficients of $\varepsilon^2 > 0$ for all v^h, and in effect we have $a\varepsilon^2 + 2b\varepsilon + c \geq 0$, therefore the roots are equal, or complex, which implies that $b^2 \leq 4ac$, but $c = 0$ in (11.55) and therefore we have that $b = 0$, which implies that $a\left(u^h, v^h\right) = \left(f, v^h\right)$ and if $S^h \to \mathcal{H}_e^1$ then $a(u, v) = (f, v)$. However, this needs $\mathcal{H}_e^1$ to be closed. The proof of this is shown in [417] that when $\mathcal{H}_B^1$, where the B indicates that all the boundary conditions are satisfied, is closed, implies that if there are a limited number of points included, then this Hilbert space becomes $\mathcal{H}_e^1$, which implies $\mathcal{H}_e^1$ is closed.

The equation, $a(u, v) = (f, v)$, is the weak form of the differential equation. Since this expression also holds for $v = v^h$, which is due to $v^h \in S^h \subset \mathcal{H}_e^1$, then we have $a\left(u, v^h\right) = \left(f, v^h\right)$.

The next step in the proof is

$$\begin{aligned} a\left(u, v^h\right) - a\left(u^h v^h\right) &= \left(f, v^h\right) - \left(f, v^h\right), \\ A\left(u - u^h, v^h\right) &= 0. \end{aligned} \tag{11.56}$$

Now we put $e^h = u - u^h$, which then makes (11.56) $a\left(e^h, v^h\right) = 0$, $\forall v^h \in S^h$. Next we have the identities

$$u - v^h = u - u^h + u^h - v^h = e^h + u^h - v^h, \tag{11.57}$$

which implies that

$$\begin{aligned} a\left(u - v^h, u - v^h\right) &= a\left(e^h + u^h - v^h, e^h + u^h - v^h\right), \\ &= a\left(e^h, e^h\right) + 2a\left(e^h, u^h - v^h\right) + a\left(u^h - v^h, u^h - v^h\right), \\ &= a\left(e^h, e^h\right) + a\left(u^h - v^h, u^h - v^h\right) \geq 0, \end{aligned} \tag{11.58}$$

and this leads to $a\left(e^h, e^h\right) \leq a\left(u^h - v^h, u^h - v^h\right)$ with equality when $u^h = v^h$. Thus u^h also minimizes $a\left(u - v^h, u - v^h\right)$, which implies that the finite element solution minimized the *energy in the error* which is equal to the error as measured by the energy norm. This leads to the following corollary.

Corollary 11.8. *Given the expression* $a\left(e^h, v^h\right) = 0$, *and if we let* $v^h = u^h$ *then we have* $a\left(e^h, u^h\right) = 0 \Rightarrow a\left(u - u^h, u^h\right) = 0 \Rightarrow a\left(u, u^h\right) = -a\left(u^h, u^h\right)$. *This leads to*

$$\begin{aligned} a\left(a^h, e^h\right) &= a\left(u - u^h, u - u^h\right), \\ &= a(u, u) - 2a\left(u, u^h\right) + a\left(u^h, u^h\right), \\ &= a(u, u) - a\left(u^h, u^h\right), \end{aligned} \tag{11.59}$$

where (11.59) *is equivalent to*

The energy in the error = The error in the energy.

Rearranging (11.59) *results in*

$$a(u,u) = a\left(u^h, u^h\right) + a\left(e^h, e^h\right), \tag{11.60}$$

which is equivalent to Pythagoras' theorem in hyperspace; see Fig. 11.12 for an illustration.

To help illustrate that we only require the Dirichlet boundary conditions to be homogeneous in Theorem 11.7, provided that

$$a(u,u) = \iint_R \left(|\nabla u|^2 + qu^2\right) dR + \int_{S_2} \alpha u^2 dS,$$
$$\text{and } (f,u) = \iint_R fu\,dR + \int_{S_2} \beta u\,dS,$$

we consider the following differential equation:

$$\begin{aligned} -\frac{d^2u}{dx^2} + u &= -2 + x + x^2, \quad 0 < x < 1, \\ u(0) &= 0, \\ \frac{du}{dx} + u &= 5 \text{ at } x = 1. \end{aligned} \tag{11.61}$$

The solution to (11.61) can easily be shown to be $u = x + x^2$. Now we need to evaluate $a(u,u)$, which is

$$a(u,u) = \int_0^1 \left(\left(\frac{du}{dx}\right)^2 + u^2\right) dx + \left[u^2\right]_{x=1} = \frac{161}{30} + 4,$$
$$(f,u) = \int_0^1 \left(x + x^2\right)\left(-2 + x + x^2\right) dx + [Su]_{x=1} = -\frac{19}{40} + 10,$$

and therefore we have $a(u,u) = (f,u)$ if the boundary condition from S_2 is included.

We can now state the following corollary, which provides more insight in how to assess the accuracy of the finite element approximation.

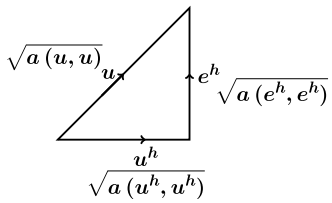


FIGURE 11.12

Pythagoras in hyperspace.

Corollary 11.9. *Since* $(u, v) = (f, u)\ \forall x \in \mathcal{H}_e^1$, *then we also have that* $a(u, w) = (f, u)$ *due to* $u \in \mathcal{H}_3^1$, *provided the term from the* S_2 *boundary is included. This implies that*

$$I[u] = a(u, u) - 2(f, u) = -a(u, u) = -(f, u). \tag{11.62}$$

In practice it is not possible to measure $a\left(e^h, e^h\right)$; however, given the theorem and the corollaries just presented, it is possible to introduce bounds on this measure. We do know that

$$a\left(e^h, e^h\right) \leq a(u - u_I, u - u_I), \tag{11.63}$$

where u_I is the interpolate of u in the solution space S^h, and is defined as

$$u_I \equiv \sum_j u_j \phi_j(x, y), \quad u_I \in S^h, \tag{11.64}$$

where u_j are the exact values at the nodes. From approximation theory we have that

$$u(a + h) = \underbrace{u(a) + hu'(a)}_{\text{Piecewise Linear}} + \frac{h^2}{2} u''(a + \theta h) \quad 0 \leq \theta \leq 1, \tag{11.65}$$

or

$$u(a + h) = \underbrace{u(a) + hu'(a) + \frac{h^2}{2} u''(a)}_{\text{PiecewiseQuadratic}} + \frac{h^2}{2} u'''(a + \theta h), \tag{11.66}$$

where the two approximations above are extendable to higher dimensions.

As an aside, it is advised to always start a practical problem by non-dimensionalizing h so that it is a fraction, where h is usually defined by the ratio of the actual size of the element to the actual size of the region.

For a linear approximation, we would expect the error in $u^h \propto h^2 u''$.

For a quadratic approximation, we would expect the error in $u^h \propto h^3 u'''$.

In general for a $k-1$-dimensional approximation, we would expect the error in $u^h \propto h^K u^{(k)}$.

Therefore, with a piecewise polynomial that is **complete** to degree $k-1$, we have the inequality

$$\|u - u_I\|_{l_2} \leq C h^k \|u^{(k)}\|_{l_2},$$

where l_2 is one of the norms defined earlier. But what do we mean by a complete polynomial? If we consider the two-dimensional case then if we had a polynomial of the form $a + bc + cy + dxy$, this polynomial is only complete for the linear terms, and hence $k = 2$. If, however, we have the polynomial $a + bc + cy + dxy + ex^2 + fy^2$, then we can see that this polynomial is complete for all of the quadratic terms then we have $k = 3$.

In Theorem 11.7 we use the energy norm that contains the squares of the first derivatives, which is defined as

$$a(v, v) = \iint_R \left(|\nabla v|^2 + qv^2\right) dR \quad \alpha = \beta = 0.$$

The first term in the integral has a dominant effect on this measure and so we have to consider how well the first derivatives are approximated. For the piecewise linear approximation we have $u'(a+h) = u'(a) + hu''(a+\theta h)$, while for the piecewise quadratic approximation we have $u'(a+h) = u'(a) + hu'(a) + \frac{h^2}{2}u'''(a+\theta h)$. This can be extended to the $k-1$ approximation in u^h, which implies that the first derivative error approximation is proportional to $h^{k-1}u^{(k)}$.

In the space where $\|u\|_{en} = \sqrt{a(u,u)}$ we have

$$a\left(e^h, e^h\right) \leq a(u - u_I, u - u_I) \leq Ch^{2(k-1)}\|u^{(k)}\|_{l_2}^2 + qCh^{2k}\|u^{(k)}\|_{l_2}^2. \tag{11.67}$$

The $h^{2(k-1)}$ term dominates the inequality as it has the lowest power of h in the expression. Hence for convergence we require $a\left(e^h, e^h\right) \to 0$ as $h \to 0$, therefore we must have k at least greater than 1, which implies we require at least a piecewise linear approximation. Now $a\left(e^h, e^h\right)$ is dominated by the squares of the first derivatives of e^h and $a\left(e^h, e^h\right)$ is $\mathcal{O}\left(h^{2(k-1)}\right)$, which implies that the error e^h is $\mathcal{O}\left(h^{2k}\right)$. Therefore we have that $\frac{e^h}{h^k} \to$ a constant as $h \to 0$. But in the finite element variational approximation, we minimize the energy norm which can be extended to a bound on

$$\|e^h\|_{l_2} = \left[\iint_R \left(e^h\right)^2 dR\right]^{1/2}, \tag{11.68}$$

through **Nitsche's trick**. A full explanation of Nitsche's trick can be found in [417].

Here we shall consider a simpler version of Nitsche's trick for linear elements and for the differential equation in problem A, where we have $Lu = f$, but with $u = 0$ on S_1, $\frac{\partial u}{\partial \mathbf{n}}$ on S_2. Now let

$$Lz = e^h,$$

where the weak form is $a(z, v) = b\left(e^h, v\right)$ for all $v \in \mathcal{H}_e^1$. We now choose $v = e^h$, which implies that

$$a\left(z, e^h\right) = b\left(e^h, e^h\right) = \|e^h\|_{l_2}^2. \tag{11.69}$$

However, from Theorem 11.7 we have

$$a\left(v^h, e^h\right) = 0, \quad \forall v^h \in S^h. \tag{11.70}$$

Subtracting (11.70) from (11.69) results in

$$a\left(z - v^h, e^h\right) = \|e^h\|_{l_2}^2 \quad \forall v^h \in S^h. \tag{11.71}$$

Before we progress we have to introduce an important inequality, which is referred to as either the Cauchy-Schwarz inequality or simply the Schwarz inequality, which states that for all vectors u and v of an inner product space, then it is true that

$$|\langle \boldsymbol{u}, \boldsymbol{v}\rangle|^2 \leq \langle \boldsymbol{u}, \boldsymbol{u}\rangle \cdot \langle \boldsymbol{v}, \boldsymbol{v}\rangle, \tag{11.72}$$

where $\langle \cdot, \cdot \rangle$ is the inner product.

Given the inequality in (11.72), it is possible to write (11.71) as

$$\|e^h\|_{l_2}^2 \leq \left(a\left(z-v^h, z-v^h\right)\right)^{1/2}\left(a\left(e^h, e^h\right)\right)^{1/2}. \tag{11.73}$$

However, we already have from approximation theory that

$$a\left(e^h, e^h\right) \leq a\left(u-u_I, u-u_I\right) \leq Ch^2\|u^{(2)}\|_{l_2}^2. \tag{11.74}$$

Also, if v^h is the finite element solution with right-hand side e^h, then we have

$$a\left(z-v^h, z-v^h\right) \leq Ch^2\|z^{(2)}\|_{l_2}^2, \tag{11.75}$$

where C is some arbitrary constant but is different from the previous use of the notation.

Now if we substitute (11.74) and (11.75) into (11.73), then we obtain

$$\|e^h\|_{l_2}^2 \leq Ch\|z^{(2)}\|_{l_2} h\|u^{(2)}\|_{l_2}. \tag{11.76}$$

However, $\|z^{(2)}\|_{l_2} \leq C\|e^h\|_{l_2}$ because $Lz = e^h$ and L contains second derivatives, hence

$$\begin{aligned} \|e^h\|_{l_2}^2 &\leq Ch^2\|u^{(2)}\|_{l_2}\|e^h\|_{l_2}, \\ \Rightarrow \|e^h\|_{l_2} &\leq Ch^2\|u^{(2)}\|_{l_2}, \end{aligned} \tag{11.77}$$

which is the expression we needed to prove.

11.4 Pin Tong

We now consider the second-order differential equation

$$-\frac{d^2u}{dx^2} = f(x), \quad 0 < x < 1, \text{ with } u(0) = a, \quad \frac{du}{dx} + \alpha u = \beta \text{ at } x = 1. \tag{11.78}$$

The finite element solution with N linear solutions of any size gives the classical solution at the nodes; this is referred to as superconvergence. This property is acceptable for any initial interval $[a, b]$ and Dirichlet at both ends, which implies that

$$u^h \equiv u_I. \tag{11.79}$$

Proof. We start by considering the weak form

$$\begin{aligned} -\int_0^1 \phi_j \frac{d^2u}{dx^2}dx &= \int_0^1 f\phi_j dx = f_j, \\ \left[\phi_j \left.\frac{\partial u}{\partial x}\right|_{x=0}\right] - \left[\phi_j\right]_{x=1}(\beta - \alpha u(1)) + \int_0^1 \frac{du}{dx}\frac{d\phi_j}{dx}dx &= f_j, \end{aligned} \tag{11.80}$$

where $j = 0, 1, \ldots, N$ but the first term in (11.80) is not initially applied at $j = 0$.

The finite element solution is determined from

$$-\left(\beta-\alpha u^n(1)\right)\left[\phi_j\right]_{x=1}+\int_0^1 \frac{du^n}{dx}\frac{d\phi_j}{dx}dx=f_j, \tag{11.81}$$

where $j=1,2,\ldots,N$, but the first term only applies for $j=N$ in (11.81)

For $j=1,2,\ldots,N$, we subtract (11.81) from (11.80), and setting $e^h=u-u^h$ results in

$$\int_0^1 \frac{de^h}{dx}\frac{d\phi_j}{dx}dx=0, \qquad j=1,2,\ldots,N-1, \tag{11.82a}$$

$$\alpha e_n+\int_0^1 \frac{de^h}{dx}\frac{d\phi_N}{dx}dx=0, \qquad j=N. \tag{11.82b}$$

From (11.82a) we have

$$\frac{1}{h_j}\int_{x_j-h_j}^{x_j}\frac{de^h}{dx}dx-\frac{1}{h_{j+1}}\int_{x_j}^{x_j+h_{j+1}}\frac{de^h}{dx}dx=0, \qquad j=1,2,\ldots,N-1,$$

$$\Rightarrow \frac{1}{h_j}\left(e_j-e_{j-1}\right)-\frac{1}{h_{j+1}}\left(e_{j+1}-e_j\right)=0, \qquad j=1,2,\ldots,N-1.$$

If we let $\lambda\equiv\frac{h_j}{h_{j+1}}$, then this implies

$$-e_{j-1}+\left(1+\lambda_j\right)e_j-\lambda_j e_{j+1}=0, \qquad j=1,2,\ldots,N-1.$$

Therefore, this gives us a recurrence relation for the errors. If we consider the equation for $j=N$, then we have

$$\alpha e_N+\frac{1}{h_N}\left(e_N-e_{N-1}\right)=0\Rightarrow e_N=\frac{1}{1-\alpha h_N}e_{N-1}.$$

Given the expression for the error at the second boundary, we can substitute the expression above into the recurrence relation for the other errors, which yields

$$\begin{aligned} e_{N-1} &= (\quad\cdots\quad)\,e_{N-2},\\ \vdots &= \qquad\vdots\\ e_2 &= (\quad\cdots\quad)\,e_1,\\ e_1 &= (\quad\cdots\quad)\,e_0, \end{aligned}$$

but $e_0=0$ as we have $u(0)=a$, where a is given. Therefore $e_j=0,\ \forall j\ 0\le j\le N$ with exact integration, where the exact integration is applied to the right-hand side. However, in practice we would apply numerical integration techniques that use interior sampling points where, for example, a **Gauss-Legendre** form of numerical integration is applied.

To help illustrate the theory presented, we again consider the torsion problem and work out why there are exact answers at the nodes. We recall the differential equation that describes this problem is

$$-\frac{\partial^2 u}{dx^2}-\frac{\partial^2 u}{\partial y^2}=2.$$

If we consider the setup in Fig. 11.13, then we can see that we have applied triangles to cover the domain.

The associated finite element equations for this problem are equivalent to a finite difference where we have

$$-\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{h_x^2}\left(-u_{i-1,j} + 2u_{i,j} - u_{i+1,j}\right). \tag{11.83}$$

Due to the symmetric formulation of the finite difference, we know that the odd derivatives in the truncation error derivation cancel, therefore the truncation error for this approximation is

$$\tau = \frac{h_x^2}{12}\frac{\partial^4 u}{\partial x^4} = 0.$$

If we now consider the y derivative, we have

$$-\frac{\partial^2 u}{\partial y^2} \approx \frac{1}{h_y^2}\left(-u_{i,j-1} + 2u_{i,j} - u_{i,j+1}\right),$$

where the truncation error associated with this approximation is

$$\tau = \frac{h_y^2}{12}\frac{\partial^4 u}{\partial y^4} = 0.$$

While it may not be obvious at first why the truncation error is equal to zero, we have to consider the fact that the classical solution for this problem is a cubic polynomial, and as such there is no fourth derivative of u. However, we should note that it is only possible to obtain exact solutions at the nodes due to the finite elements being equal triangles that are orientated the same way.

As an aside and as a word of caution, when applying different triangles to create the mesh for your differential equations it is recommended that the angles in the triangle should be at least 30 degrees due to the stiffness matrix containing the cotangent of these angles, which tend to infinity as the angles become smaller. This will make the associated stiffness matrix ill-conditioned and tending to a singular matrix, which can cause problems for inversion subroutines.

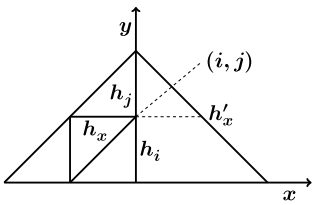


FIGURE 11.13

Triangle elements for the torsion problem domain.

11.5 Finite Element Basis Functions

We have shown that the finite element approximation is given by

$$u^h = \sum_j u_j^h \phi_j, \text{ or } u^h = \sum_j u_j^h(t)\,\phi_j,$$

and we always have the conditions on the basis functions

$$\phi_j = \begin{cases} 1 & \text{at node } j \\ \\ 0 & \text{at all other nodes in the neighborhood of } j \end{cases}$$

$\phi_j \equiv 0$ outside the **neighborhood** of node j.

We now consider some different candidates for basis functions for different dimensions.

11.5.1 One Dimension

For the one-dimensional case we only require the basis functions to be C^0, which as we know is the set of functions that are continuous. The simple linear approximation for this setup is shown in Fig. 11.14, where we can see that we have two nodes and therefore we have two degrees of freedom, and we have the global coordinate x. We introduce the local coordinate ξ which is given by $x = x_j + \xi h_j$, with $0 \leq \xi \leq 1$. This then makes the basis function as $\phi_j = 1 - \xi$, $\phi_{j+1} = \xi$.

For a quadratic approximation we have three degrees of freedom. We then require an extra node and as such this extra node will always be at the midpoint. We have highlighted how the basis functions could look for the quadratic-based basis functions in Fig. 11.15.

Exercise 11.10. *Show that the three quadratic basis functions on the interval* $[0, 1]$*, where there are nodes at* 0*,* 0.5*, and* 1 *are,*

$$\phi_0 = 1 - 3x + 2x^2, \qquad \phi_{\frac{1}{2}} = 4\left(x - x^2\right), \qquad \phi_1 = 2x^2 - x.$$

The local coordinate system is introduced through the transformation $\xi = \frac{2}{h}\left(x - \left(x_j + \frac{h}{2}\right)\right)$. This makes the basis function

$$\phi_{-1} = -\frac{\xi\,(1-\xi)}{2}, \qquad \phi_0 = 1 - \xi^2, \qquad \phi_1 \frac{\xi\,(1-\xi)}{2}.$$

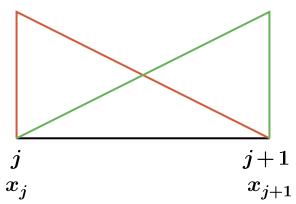


FIGURE 11.14

Linear basis function diagram.

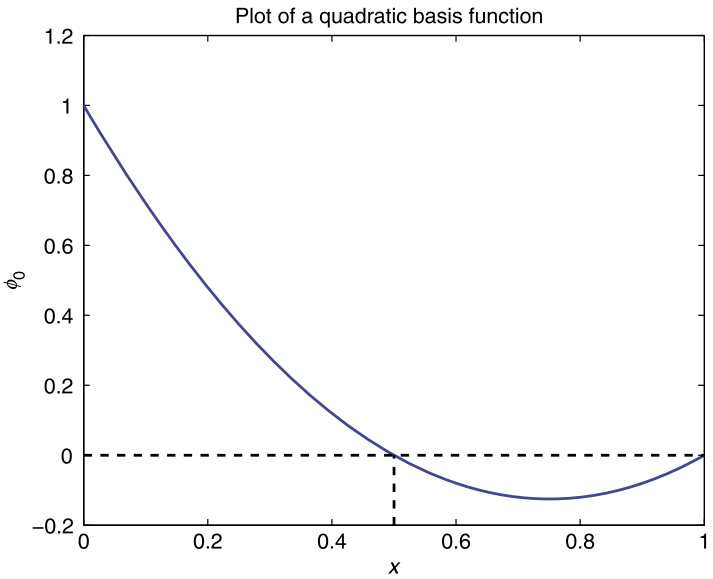


FIGURE 11.15

Plot of the quadratic basis function ϕ_0.

The expressions above may look familiar. They should if you have read the semi-Lagrangian chapter, as they are the coefficients of the quadratic Lagrange interpolation polynomial.
Note: For the element stiffness matrix, we require

$$\int_{x_j}^{x_j+h} \frac{d\phi_{-1}}{dx}\frac{d\phi_0}{dx}dx \equiv \int_{-1}^{1} \frac{d\phi_{-1}}{d\xi}\frac{d\xi}{dx}\frac{d\phi_0}{d\xi}\frac{d\xi}{dx}\frac{dx}{d\xi}d\xi,$$
$$= \frac{2}{h}\int_{-1}^{1}\frac{1}{2}(1-2\xi)\,2\xi d\xi = -\frac{8}{3h}.$$

Exercise 11.11. *Show that the term* $\int_{x_j}^{x_j+h}\left(\frac{d\phi_{-1}}{dx}\right)^2 dx = \frac{7}{3h}$.
This leads to the stiffness matrix

$$\frac{1}{3h}\begin{pmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{pmatrix},$$

which as always must be singular.

11.5.2 Two Dimensions

If we consider the linear case first, where we are using triangles, then we have three degrees of freedom, with three nodal values that determine a plane polyhedral surface that will be C^0. For quadratic triangles

we require a complete quadratic polynomial that involves x, y, x^2, xy, y^2, which implies that we again require extra nodes at the midpoints. We have provided the triangles for the linear and quadratic triangles in Fig. 11.16.

We then have C^1 for a shell or a plate. To ensure that we have C^1 triangles requires 21 degrees of freedom.

11.6 Coding Finite Element Approximations for Triangle Elements

Given all the theory that we have introduced in the previous sections, we now consider how to implement these approaches into computer codes. We start by introducing the **local** coordinates which are referred to as **area** coordinates. We consider the situation presented in Fig. 11.17 where we have the general point P that has area coordinates L_1, L_2, L_3 where we have $L_1 \frac{\triangle P23}{\triangle 123}$.

Each $L_j = 1$ at node j and 0 at the other nodes, so we put $\phi_j = L_j$ for linear triangles. We should note that $L_1 + L_2 + L_3 = \frac{\triangle 123}{\triangle 123} = 1$.

The area of the triangle with vertices (x_j, y_j), $j = 1, 2, 3$ are given by

$$\triangle_j = \pm\frac{1}{2}\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix},$$

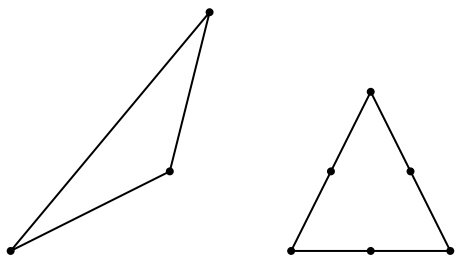


FIGURE 11.16

Linear and quadratic triangular elements.

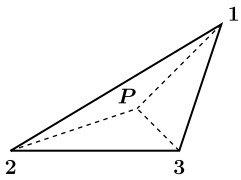


FIGURE 11.17

Local/area coordinate schematic.

where the area would be positive if vertices are counter-clockwise, and we have that (x, y) are the global coordinates.

The area coordinates local to the triangle $\triangle 123$ are given by the *local global* transformation, defined as

$$L_1 = \frac{1}{2\Delta_c}\begin{vmatrix} 1 & x & y \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \Rightarrow L_1 = \frac{1}{2\Delta_c}\left((x_2 y_3 - x_3 y_2) - x\,(y_3 - y_2) + y\,(x_3 - x_2)\right),$$
$$= \frac{1}{2\Delta_c}\left(b_1 + c_1 x + d_1 y\right),$$

where the coefficients b_1, c_1, and d_1 are the cofactors of the cyclic rotation.

If we consider the problem

$$-\nabla^2 u = 4, \text{ on } x^2 + y^2 < 1, \text{ with } u = 0 \text{ on } x^2 + y^2 = 1,$$

then the solution to this problem is $u = 1 + x^2 + y^2 = 1 - r^2$.

We only require the stiffness matrix to solve this problem with a finite element approaches, where we shall use linear triangles. There are two different setups that we can consider; the first is referred to as type A and is illustrated in Fig. 11.18.

With this configuration there is only a slope in the x-direction and that we have a series of N concentric circles with radii such that $0 = r_N < r_{N-1} < \cdots < r_2 < r_1 = 1$. We denote the finite element approximation to the solution by v, which is given by

$$v = q_{i-1}\frac{r_{i-1}\cos\alpha - x}{r_{i-1}\cos\alpha - r_i} = \begin{cases} 1 & \text{at } x = r_i, \\ 0 & \text{at } x = r_i\cos\alpha, \end{cases}$$

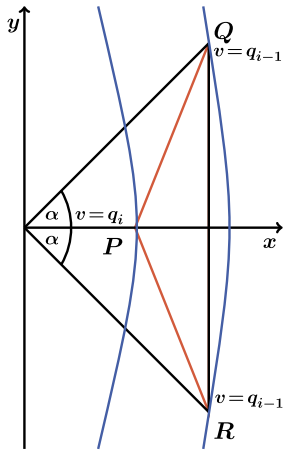


FIGURE 11.18

Configuration for the A triangles.

which leads to the finite element approximation being given by

$$v = q_i \frac{r_{i-1}\cos\alpha - x}{r_{i-1}\cos\alpha - r_i} + q_{i-1}\frac{x - r_i}{r_{i-1}\cos\alpha - r_i} = q_i\phi_P + q_{i-1}\left(\phi_Q + \phi_R\right). \tag{11.84}$$

The contribution to the $\mathbf{K}_q$ matrix is of the form

$$\iint_{\triangle PQR} \left(q_i \nabla\phi_P + q_{i-1}\left(\phi_Q - \phi_R\right)\right)\cdot\nabla\phi_R dxdy, \tag{11.85}$$

where $k = P, Q, R$. When $k = P$ we have $q_i \iint_{\triangle PQR}\nabla\phi_P\nabla\phi_P dR + q_{i-1}\iint_{\triangle PQR}\nabla\left(\phi_Q + \phi_R\right)\cdot\nabla\phi_P dR$, which is equivalent to

$$q_i\frac{\triangle}{\left(r_{i-1}\cos\alpha - r_i\right)^2} + q_{i-1}\frac{(-1)\triangle}{\left(r_{i-1}\cos\alpha - r_i\right)^2},$$

where $\triangle \equiv r_{i-1}\sin\alpha\left(r_{i-1}\cos\alpha - r_i\right)$. Given this expression for the area of the triangle, we see that the formula above simplifies to

$$q_i\frac{r_{i-1}\sin\alpha}{r_{i-1}\cos\alpha - r_i} - q_i\frac{r_{i-1}\sin\alpha}{r_{i-1}\cos\alpha - r_i} \equiv q_i c_i - q_{i-1}c_i,$$

which is what is required as we have the condition that element stiffness matrix must initially be singular.

The second possible configuration for the elements for this problem, which we shall refer to as type B, is still based upon the triangle as the element being applied, but now configured in a different direction. We have presented a diagram of the triangular configuration for the type B problem in Fig. 11.19.

Therefore, following the same arguments from the type A configuration, we have that our finite element approximation v is given by

$$v = q_{i-1}\frac{(x - r_i\cos\alpha)}{r_{i-1} - r_i\cos\alpha} + q_i\frac{\left(r_{i-1} - x\right)}{r_{i-1} - r_i\cos\alpha}. \tag{11.86}$$

Applying the same integral to the finite element approximation in (11.86), results in the coefficient for the stiffness matrix as

$$c_i' \equiv \frac{r_i\sin\alpha}{r_{i-1} - r_i\cos\alpha}.$$

Combining the two types of triangles for their values at each node results in the general expression for entries of the stiffness matrix as

$$\mathbf{K}\boldsymbol{q} \equiv \begin{pmatrix} C_1 & -C_1 & 0 & \cdots & \cdots & 0 \\ -C_1 & C_1 + C_2 & -C_2 & 0 & \cdots & 0 \\ 0 & -C_2 & C_2 + C_3 & -C_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \cdots & \vdots \\ 0 & \cdots & \cdots & C_{N-1} & C_{N-1} + C_N & -C_N \\ 0 & \cdots & \cdots & \cdots & -C_{N-1} & C_N \end{pmatrix}\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ \vdots \\ q_{N-1} \\ q_N \end{pmatrix},$$

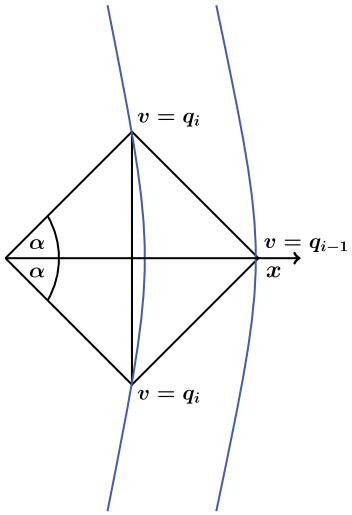


FIGURE 11.19

Configuration for the B triangles.

where $C_i = M\left(c_i + c_i'\right)$, and where on the inner most circle we only have type A triangles and where the stiffness matrix for this circle becomes $C_N = M\tan\alpha$. However, we can remove the first row and column due to $q_0 = 0$ on the boundary. This them leaves an Nth-order tridiagonal matrix to invert.

11.6.1 Square Elements

So far we have considered different triangle approximations for the grid that we are using for the finite element approximation; however, it is possible to use many different polygons to form the grid. The first non-triangular grid we consider is a square-based grid. If we consider a square whose corners are at $(-1,-1)$, $(1,-1)$, $(1,1)$, and $(-1,1)$ then we can have a four node bilinear approximation to our differential equation with four degrees of freedom. As before, we shall introduce the local coordinates η and ξ where $-1 \le \eta \le 1$ and $-1 \le \xi \le 1$.

The local coordinates are aligned with the global coordinates x, y axes and are related to each other by

$$x = \overline{x} + \frac{h}{2}\xi,$$
$$y = \overline{y} + \frac{h}{2}\eta,$$

where $(\overline{x}, \overline{y})$ is the center of the square. Note that the two expressions above can be inverted to find expressions for the local coordinates as

$$\xi = \frac{2}{h}\left(x - \overline{x}\right),$$
$$\eta = \frac{2}{h}\left(y - \overline{y}\right).$$

Given the setup above, it is possible to define the four bilinear basis functions as

$$\phi_1 = \frac{1}{4}(1-\xi)(1-\eta),$$
$$\phi_2 = \frac{1}{4}(1+\xi)(1-\eta),$$
$$\phi_3 = \frac{1}{4}(1+\xi)(1+\eta),$$
$$\phi_4 = \frac{1}{4}(1-\xi)(1+\eta),$$

where each basis function is bilinear due to the product of two linear brackets and result in functions of 1, ξ, η, and $\xi\eta$.

If we now consider the element stiffness matrix, then we have

$$K_{i,j}^{(e)} = \iint_e \left(\frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y}\frac{\partial \phi_j}{\partial y} \right) dxdy, \tag{11.87}$$

and we have to convert this expression into the local coordinates via the chain rule, which yields

$$\frac{\partial \phi_i}{\partial x} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial \phi_i}{\partial \eta}\frac{\partial \eta}{\partial x} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial x},$$
$$\frac{\partial \phi_i}{\partial y} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial y} + \frac{\partial \phi_i}{\partial \eta}\frac{\partial \eta}{\partial y} = \frac{\partial \phi_i}{\partial \eta}\frac{\partial \eta}{\partial y},$$

where the associated determinant of the Jacobian for this change of variable is given by

$$|J| = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \begin{vmatrix} \frac{h}{2} & 0 \\ 0 & \frac{h}{2} \end{vmatrix} = \frac{h^2}{4}. \tag{11.88}$$

This enables us to write (11.87) in terms of the local coordinates as

$$\begin{aligned} K_{i,j}^{(e)} &= \iint_e \left(\frac{\partial \phi_i}{\partial \xi}\frac{2}{h}\frac{\partial \phi_j}{\partial \xi}\frac{2}{h} + \frac{\partial \phi_i}{\partial \eta}\frac{2}{h}\frac{\partial \phi_j}{\partial \eta}\frac{2}{h} \right) \frac{h^2}{4} d\xi d\eta, \\ &= \iint_e \left(\frac{\partial \phi_i}{\partial \xi}\frac{\partial \phi_j}{\partial \xi} + \frac{\partial \phi_i}{\partial \eta}\frac{\partial \phi_j}{\partial \eta} \right) d\xi d\eta, \\ &= \int_{-1}^{1}\int_{-1}^{1} \left(\frac{\partial \phi_i}{\partial \xi}\frac{\partial \phi_j}{\partial \xi} + \frac{\partial \phi_i}{\partial \eta}\frac{\partial \phi_j}{\partial \eta} \right) d\xi d\eta, \end{aligned}$$

whereupon applying the integration above results in

$$\mathbf{K}^e = \frac{1}{6}\begin{pmatrix} 4 & -1 & -2 & -1 \\ -1 & 4 & -1 & -2 \\ -2 & -1 & 4 & -1 \\ -1 & -2 & -1 & 4 \end{pmatrix}, \tag{11.89}$$

which is symmetric and singular.

We now consider the following example to illustrate how to incorporate the mixed boundary conditions $\frac{\partial u}{\partial \boldsymbol{n}} + \alpha u = \beta$ in a two-dimensional region. We shall suppose that we have **one** bilinear element to

solve $-\nabla u = f$ on the unit square, which implies that the sides are of length 1. To help with keeping track of the numbering of the nodes for this problem, we have provided a simple diagram in Fig. 11.20.

For this example we have the following boundary conditions: on the boundary between nodes 1 and 2 we have $u = a$, on the boundaries between nodes 2 and 3, as well as on the boundary between nodes 1 and 4 we have that the outward normal derivative is zero, $\frac{\partial u}{\partial \boldsymbol{n}} = 0$; finally on the boundary between the nodes 3 and 4 we have the mixed boundary condition, $\frac{\partial u}{\partial \boldsymbol{n}} + \alpha u = \beta$. As we have seen before, the α term will add an extra term to the left-hand side of the matrix equation and the β term will add an extra term to the right-hand side of the matrix equation.

If we consider the weak form approach, then we introduce $w = \phi_k$ as a basis function and integrate by parts, which yields

$$
\begin{aligned}
-\iint_R \phi_k \nabla^2 u \, dR &= \iint_R f \phi_k \, dR, \\
-\int_{S_1+S_2} \phi_k \frac{\partial u}{\partial \boldsymbol{n}} dS + \iint_R \nabla \phi_k \cdot \nabla u \, dR &= \iint_R f \phi_k \, dR, \\
-\int_{12} \phi_k \frac{\partial u}{\partial \boldsymbol{n}} dS - \int_{34} \phi_k \left(\beta - \alpha u\right) dS + \iint_R \left(\nabla \phi_k \cdot \nabla \sum_j u_j^h \phi_j \right) dR &= f_k,
\end{aligned}
\tag{11.90}
$$

where the boundary conditions between nodes 23 and 14 have enabled us to eliminate that part of the integral around the boundary. However, we do not know the expression for the outward normal between nodes 12, and so we only use the basis functions ϕ_3 and ϕ_4, that are zero between 12, but these will enable us to obtain the two unknowns u_3^h and u_4^h. To achieve this goal we introduce the local coordinates $O\xi$ and $O\eta$, where O is the center of the square.

On the boundary between nodes 34 we have $\eta = 1$ and $u^h = u_4^h \frac{(1-\xi)}{2} + u_3^h \frac{(1+\xi)}{2}$ along with $dS = -\frac{1}{2} d\xi$, and so we integrate from $\xi = 1$ to $\xi = -1$ to obtain the negative direction. Therefore, for $k = 3$ we have

$$
-\frac{\alpha}{2} \int_1^{-1} \left(\frac{1+\xi}{2} \right) \left(u_4^h \left(\frac{1-\xi}{2} \right) + u_3^h \left(\frac{1+\xi}{2} \right) \right) d\xi + \mathbf{K} = f_k - \frac{\beta}{2} \int_1^{-1} \left(\frac{1+\xi}{2} \right) d\xi,
\tag{11.91}
$$

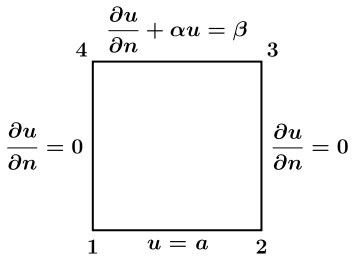


FIGURE 11.20

Square domain example.

which results in

$$\frac{\alpha}{3}u_3^h + \frac{\alpha}{6}u_4^h + \left(-\frac{1}{3}u_1^h - \frac{1}{6}u_2^h + \frac{2}{3}u_3^h - \frac{1}{6}u_4^h\right) = f_k + \frac{\beta}{2}, \tag{11.92}$$

where we have used the third row of the element stiffness matrix we derived earlier and where we have also used the property that

$$\int_1^{-1} \xi^{2n+1} d\xi = 0, \text{ and } \int_1^{-1} \xi^{2n} d\xi = \frac{2}{2n+1}.$$

Therefore, we can write (11.92) as

$$\frac{1}{3}(\alpha + 2)\, u_3^h + \frac{1}{6}(\alpha - 1)\, u_4^h = f_k + \frac{\beta}{2} + \frac{1}{3}u_1^h + \frac{1}{6}u_2^h. \tag{11.93}$$

It is possible to obtain a similar expression for $k = 4$ to that in (11.93), which results in two equations in two unknowns as we know that $u_1^h = u_2^h = a$ from the boundary condition.

Returning to (11.90) and if we set $\phi_k = \phi_1$ and ϕ_2, then we obtain the equations

$$-\int_{12} \phi_1 \frac{\partial u}{\partial \boldsymbol{n}} dS + \iint_R \nabla\phi_1 \cdot \nabla u^h dR = f_1, \tag{11.94a}$$

$$-\int_{12} \phi_2 \frac{\partial u}{\partial \boldsymbol{n}} dS + \iint_R \nabla\phi_2 \cdot \nabla u^h dR = f_2. \tag{11.94b}$$

Adding (11.94a) and (11.94b) results in

$$\int_{12} (\phi_1 + \phi_2) \frac{\partial u}{\partial \boldsymbol{n}} dS + \iint_R (\nabla\phi_1 + \nabla\phi_2) \cdot \nabla u^h dR = f_1 + f_2.$$

However, on the boundary between nodes 1 and 2 the sum of the two basis functions ϕ_1 and ϕ_2 is equal to one. Now recall that the basis functions are one at their nodes, and decay linearly to the adjacent nodes. Therefore $\phi_1 + \phi_2 = 1$ between 12. This then enables us to find an expression for the $-\int_{12} \frac{\partial u}{\partial \boldsymbol{n}}$ term.

11.7 Isoparametric Elements

We now consider the situation where we wish to use a standard polynomial element: for example, triangles or rectangles. If we have subdivided a region that was not of the proper shape, then it could be the case that we have one of more curved sides, or they may be non-rectangular quadrilaterals. By changing to a new $\xi - \eta$ coordinate system, the elements can be given the correct shape. The element stiffness matrices are then evaluated by integrations in the new variables over triangles or rectangles, and minimization then leads to the finite element solution, $u^h(\xi, \eta)$, which can be transformed back to x to y. We have drawn a simple schematic of the change of the isoparametric element to a square element in Fig. 11.21.

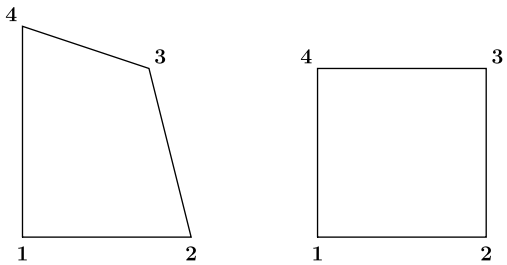


FIGURE 11.21

Simple illustration of an isoparametric transformation for a quadrilateral to a square.

We have that $x = \sum_1^4 x_j \phi_j(\xi, \eta)$, $y = \sum_1^4 \phi_i(\xi, \eta)$, therefore we can now form the element stiffness matrix, where

$$\iint_{element} \frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x} + \frac{\partial \phi_i}{\partial y}\frac{\partial \phi_j}{\partial y} dx dy, \tag{11.95}$$

with

$$\frac{\partial \phi_i}{\partial x} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial \phi}{\partial \eta}\frac{\partial \eta}{\partial x}. \tag{11.96}$$

We also have that

$$\frac{\partial \phi_i}{\partial \xi} = \frac{\partial \phi}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial \phi_i}{\partial y}\frac{\partial y}{\partial \xi}, \tag{11.97}$$

$$\frac{\partial \phi_i}{\partial \eta} = \frac{\partial \phi}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial \phi_i}{\partial y}\frac{\partial y}{\partial \eta}. \tag{11.98}$$

This can be written in matrix form as

$$\begin{pmatrix} \frac{\partial \phi_i}{\partial \eta} \\ \frac{\partial \phi_i}{\partial \eta} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} \begin{pmatrix} \frac{\partial \phi_i}{\partial x} \\ \frac{\partial \phi_i}{\partial y} \end{pmatrix} \Rightarrow \begin{pmatrix} \frac{\partial \phi_i}{\partial x} \\ \frac{\partial \phi_i}{\partial y} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial \phi_i}{\partial \eta} \\ \frac{\partial \phi_i}{\partial \eta} \end{pmatrix}. \tag{11.99}$$

Therefore we have what are referred to as shape functions for the bilinear quadrilateral shape that we are using here, defined as

$$\begin{aligned} \phi_1 &= \tfrac{1}{4}(1-\xi)(1-\eta), \quad \phi_2 = \tfrac{1}{4}(1+\xi)(1-\eta), \\ \phi_3 &= \tfrac{1}{4}(1+\xi)(1+\eta), \quad \phi_4 = \tfrac{1}{4}(1-\xi)(1+\eta). \end{aligned} \tag{11.100}$$

We also have what are referred to as the **displacement interpolations**, that are denoted by u_x and u_y such that we can form the vector matrix equation

$$\begin{pmatrix} 1 \\ x \\ y \\ u_x \\ u_y \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ u_{x1} & u_{x2} & u_{x3} & u_{x4} \\ u_{y1} & u_{y2} & u_{y3} & u_{y4} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix}. \tag{11.101}$$

This then implies

$$\begin{aligned} x &= x_1\phi_1 + x_2\phi_2 + x_3\phi_3 + x_4\phi_4, \\ &= x_1\frac{1}{4}(1-\xi)(1-\eta) + x_2\frac{1}{4}(1+\xi)(1-\eta) + x_3\frac{1}{4}(1+\xi)(1+\eta) + x_4\frac{1}{4}(1-\xi)(1+\eta), \\ y &= y_1\phi_1 + y_2\phi_2 + y_3\phi_3 + y_4\phi_4, \\ &= y_1\frac{1}{4}(1-\xi)(1-\eta) + y_2\frac{1}{4}(1+\xi)(1-\eta) + y_3\frac{1}{4}(1+\xi)(1+\eta) + y_4\frac{1}{4}(1-\xi)(1+\eta). \end{aligned}$$

Exercise 11.12. *Given the expressions above for the coordinate change, show that*

$$\begin{aligned} \frac{\partial x}{\partial \xi} &= \frac{1}{4}(x_1 - x_2 - x_3 + x_4 + \eta(x_1 - x_2 + x_3 - x_4)), \\ \frac{\partial y}{\partial \eta} &= \frac{1}{4}(-y_1 - y_2 + y_3 + y_4 + \xi(y_1 - y_2 + y_3 - y_4)). \end{aligned}$$

Verify that the expression for this transform is linear in ξ, η by verifying that $|J|$ is independent of ξ, η.

An important property to check here is that $|J|$ cannot be zero in the interior of the element, because this would imply a functional dependence between x and y. If we consider the determinant of the Jacobian at $\xi = \eta = 1$, then we have

$$|J|_{\xi=\eta=1} = \frac{1}{4}((x_1 - x_2)(y_1 - y_4) - (y_1 - y_2)(x_1 - x_4)). \tag{11.102}$$

Therefore, by the cross product formula, this expression is equal to $\frac{ll'}{4}\sin\theta$, where for $\theta < \pi$ we have that $|J|$ will remain positive (see Fig. 11.22).

If we now consider the case of an isoparametric transformation where we have eight nodes of a curved, non-regular, shape, then as we can see from Fig. 11.23, it is possible to transform this shape to

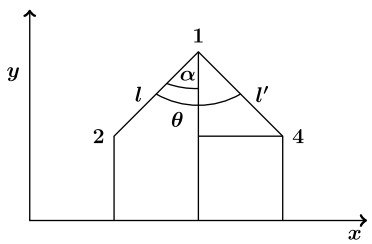


FIGURE 11.22

Schematic of the angles involved in the isoparametric transforms.

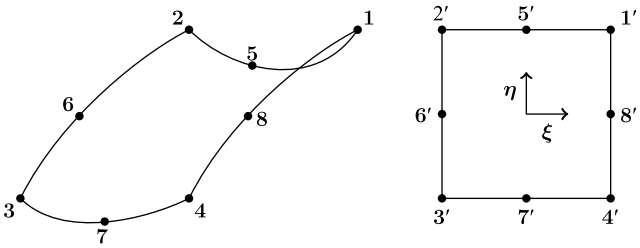


FIGURE 11.23

Simple illustration of an eight-node isoparametric transformation for a quadrilateral to a square.

a regular square with the expressions

$$x = \sum_{i=1}^{8} x_i \phi_i(\xi, \eta), \quad y = \sum_{i=1}^{8} y_i \phi_i(\xi, \eta).$$

Therefore the basis functions, and the interpolation/transform polynomial are biquadratic, and this problem is referred to as the **biquadratic quadrilateral** problem, and is given by

$$\phi_1 = \frac{1}{4}(1-\xi)(1-\eta)\xi\eta, \qquad \phi_5 = -\frac{1}{2}\left(1-\xi^2\right)(1-\eta)\eta, \tag{11.103a}$$

$$\phi_2 = -\frac{1}{4}(1+\xi)(1-\eta)\xi\eta, \qquad \phi_6 = \frac{1}{2}(1+\xi)\left(1-\eta^2\right)\xi, \tag{11.103b}$$

$$\phi_3 = \frac{1}{4}(1+\xi)(1+\eta)\xi\eta, \qquad \phi_7 = \frac{1}{2}\left(1-\xi^2\right)(1+\eta)\eta, \tag{11.103c}$$

$$\phi_4 = -\frac{1}{4}(1-\xi)(1+\eta)\xi\eta, \qquad \phi_8 = -\frac{1}{2}(1-\xi)\left(1-\eta^2\right)\xi. \tag{11.103d}$$

The element above are also referred to as the **Lagrangian quadrilateral**. This element is useful with curved sides on a curved boundary.

Exercise 11.13. *Verify that the basis function in* (11.103d) *satisfy the conditions of being equal to one at* i *and equal to zero at neighboring point and equal to zero for all other points in the domain.*

The last isoparametric element that we consider is the six-node triangle. This is the element where we have a curved triangle and we transform the elements into a right-angled triangle in the (ξ, η) coordinate system. This element is very useful with a curved boundary approximated by a section of a parabola. Fig. 11.24 shows a schematic.

For details of the exact transformation between the two coordinate systems for this situations $(x, y) \mapsto (\xi, \eta)$, see Strang and Fix [417], or a later edition. However, we can say that when we are in the local coordinates, then we use a biquadratic formulation for our basis functions of the form

$$\phi = a_1 + a_2\xi + a_3\eta + a_4\xi^2 + a_5\xi\eta + a_6\eta^2. \tag{11.104}$$

If we take point 1, which is at the origin of the local coordinates, then we know that this function must be equal to one here, decrease to zero at the neighboring points, and be zero at the other nodes.

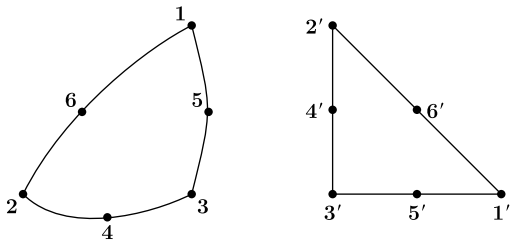


FIGURE 11.24

Simple illustration of a six-node isoparametric transformation for a distorted triangle to a regular triangle.

Given these conditions, we can develop set of six simultaneous equations in six unknowns at the six points. It can easily be shown that the basis function at this point is given by

$$\phi_1 = 1 - 3\xi - 3\eta + 2\xi^2 + 2\eta^2 + 4\xi\eta. \tag{11.105}$$

In Fig. 11.25 we have plotted the surface that the polynomial in (11.105) defines to show that in the new coordinates this function satisfies the requirements to be a basis function for this problem.

Exercise 11.14. *Find the basis functions for the other five points on the triangle.*

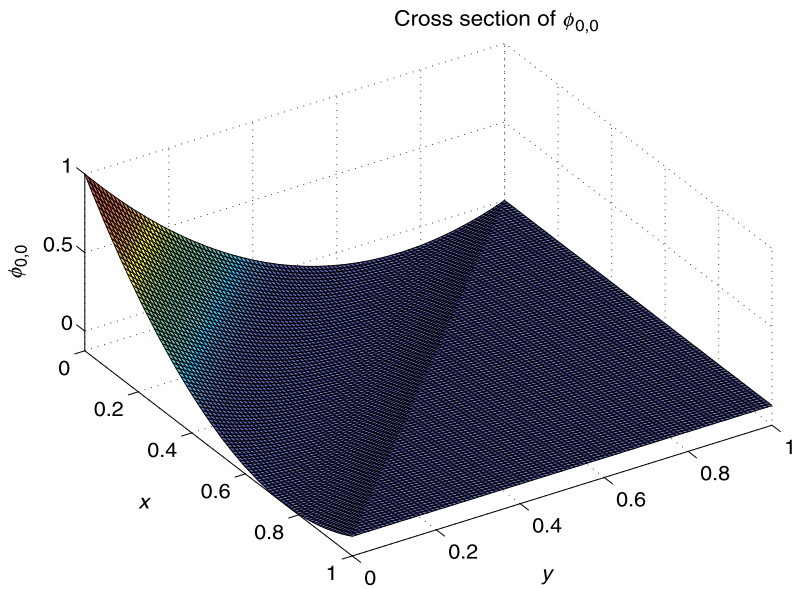


FIGURE 11.25

Surface plot of the biquadratic basis function for $\phi_{0,0}$.

11.8 Summary

We have only scratched the surface of FEM in this chapter, but we wanted to introduce the reader to at least the basics of FEM to have some understanding of how this theory is applied. We have introduced the functional and the weak form of finite element modeling. We have shown that this leads to the mass and stiffness matrices to invert to find the numerical solution to the discrete approximation to the differential equations. We have introduced the error analysis for these schemes, as we did for the finite difference schemes, where for FEM this was achieved through the use of *Nitsche's trick*; see Strang and Fix [417] for more details about this *trick*. We have also introduced the basis functions which play a vital part in the application of FEM.

We have extended the FEM theory into two dimensions, where now we have shape, or more specifically elements, that are used to form the grid over the domains of interest. We have introduced the theory for triangular and square elements, where we can have either a bilinear or biquadratic basis functions. The final part of the finite element theory introduced in this chapter was associated with the isoparametric shapes. These shapes were seen to be distorted triangle or quadrilaterals, but where through a correct change of coordinate system from the global, (x, y), to the local coordinates, (ξ, η), it was possible to obtain the regular shapes. As such we could perform the finite element analysis in this coordinate system, and invert the solution back into the global coordinate system.

We now move on from modeling in Cartesian coordinates to consider the problem of numerically approximating differential equations in spherical coordinates.

CHAPTER

Numerical Modeling on the Sphere

12

Contents

Spherical modeling plays a vital part in atmospheric as well as ocean numerical modeling. There are several differences between Cartesian coordinate based modeling and spherical coordinate based modeling. In this chapter shall present the differences between the two coordinate systems and their impacts on differential operators, as well as projections, types of grids to implement the numerical approximations on, Fourier analysis, as well as the components of spectral modeling. We start with the changes to the vector operators.

12.1 Vector Operators in Spherical Coordinates

When transforming from Cartesian coordinates to spherical coordinates, the direction in which the unit vectors are pointing changes and this has to be taken into account in the vector operators. Therefore, we now consider how the definitions for the unit vectors are different in spherical coordinates compared to Cartesian.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00016-4

12.1.1 Spherical Unit Vectors

In spherical coordinates there is a local approximation through a tangential plane relative to the spherical surface, where the unit vectors in spherical coordinates are defined. See Fig. 12.1 for an illustration of the tangential plane to the sphere, where θ is the angle of colatitude, defined in $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, λ is the angle of longitude, defined in $\lambda \in [0, 2\pi)$, and r is the radial distance defined in $r \in [0, R]$, where R is a set distance from the center of the sphere.

It is from this tangential plane approximation that the following nine derivatives for the unit vectors in a 3D framework arise:

$$\frac{\partial \mathbf{i}}{\partial r} = 0, \qquad \frac{\partial \mathbf{i}}{\partial \theta} = 0, \qquad \frac{\partial \mathbf{i}}{\partial \lambda} = -\cos\theta \mathbf{k} + \sin\theta \mathbf{j}, \tag{12.1}$$

$$\frac{\partial \mathbf{j}}{\partial r} = 0, \qquad \frac{\partial \mathbf{j}}{\partial \theta} = -\mathbf{k}, \qquad \frac{\partial \mathbf{j}}{\partial \lambda} = -\sin\theta \mathbf{i}, \tag{12.2}$$

$$\frac{\partial \mathbf{k}}{\partial r} = 0, \qquad \frac{\partial \mathbf{k}}{\partial \theta} = \mathbf{j}, \qquad \frac{\partial \mathbf{k}}{\partial \lambda} = \cos\theta \mathbf{i}. \tag{12.3}$$

The full derivation of these expressions can be found in [26]. If we consider a 2D framework, this implies that there is a constant radial distance and hence there is no change along the radial, $\mathbf{k}$, direction. Thus all the terms in the derivatives that contain $\mathbf{k}$ are equal to zero, leaving

$$\frac{\partial \mathbf{i}}{\partial \lambda} = \sin\theta \mathbf{j}, \qquad \frac{\partial \mathbf{j}}{\partial \lambda} = -\sin\theta \mathbf{i}. \tag{12.4}$$

12.2 Spherical Vector Derivative Operators

We now consider how to alter the Cartesian definition for the gradient operator ∇, the divergence operator $\nabla\cdot$, the curl operator $\nabla\times$, the Laplacian operator ∇^2, and finally the Jacobian into spherical coordinates. We start by defining the expression for the derivatives of a vector $\mathbf{G}$, where for

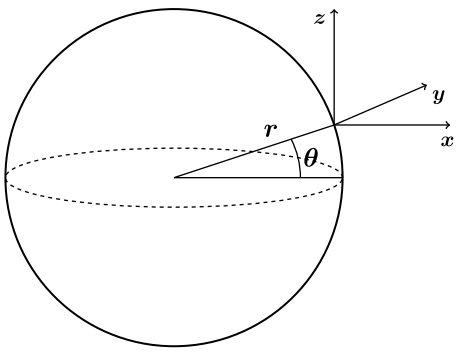


FIGURE 12.1

Diagram of the tangential coordinates on a spherical surface, where $\boldsymbol{r}$ is the radius of the sphere, and the distance laid out from the radius is $\boldsymbol{\theta}$.

the Cartesian coordinates we have $\boldsymbol{G} = \left(G_x, G_y, G_z\right)^T$, while for the spherical coordinates we have $\boldsymbol{G} = (G_\lambda, G_\theta, G_r)^T$.

The definition of the gradient operator of a scalar function F is given by

$$\nabla F = \mathbf{i}\frac{\partial F}{\partial x} + \mathbf{j}\frac{\partial F}{\partial y} + \mathbf{k}\frac{\partial F}{\partial z},$$

which in spherical coordinates is

$$\nabla F = \frac{\mathbf{i}}{r\cos\theta}\frac{\partial F}{\partial \lambda} + \frac{\mathbf{j}}{r}\frac{\partial F}{\partial \theta} + \mathbf{k}\frac{\partial F}{\partial r}.$$

The next operator is the divergence, which for the vector field $\mathbf{G}$ is defined in Cartesian coordinates as

$$\nabla \cdot \mathbf{G} = \frac{\partial G_x}{\partial x} + \frac{\partial G_y}{\partial y} + \frac{\partial G_z}{\partial z},$$

and in spherical coordinates by

$$\nabla \cdot \mathbf{G} = \frac{1}{r^2\cos\theta}\left\{\frac{\partial r G_\lambda}{\partial \lambda} + \frac{\partial r\cos\theta G_\phi}{\partial \theta} + \frac{\partial r^2\cos\theta G_r}{\partial r}\right\}.$$

The curl operator in Cartesian coordinates is defined as

$$\nabla \times \mathbf{G} = \left(\frac{\partial G_z}{\partial y} - \frac{\partial G_y}{\partial z}\right)\mathbf{i} - \left(\frac{\partial G_z}{\partial x} - \frac{\partial G_x}{\partial z}\right)\mathbf{j} + \left(\frac{\partial G_y}{\partial x} - \frac{\partial G_x}{\partial y}\right)\mathbf{k},$$

whereas in spherical coordinates it is defined as

$$\nabla \times \mathbf{G} = \frac{\mathbf{i}}{r}\left(\frac{\partial G_r}{\partial \theta} - \frac{\partial r G_\theta}{\partial r}\right) + \frac{\mathbf{j}}{r\cos\theta}\left(\cos\theta\frac{\partial r G_\lambda}{\partial r} - \frac{\partial G_r}{\partial \lambda}\right) + \frac{\mathbf{k}}{r\cos\theta}\left(\frac{\partial G_\theta}{\partial \lambda} - \frac{\partial \cos\theta G_\lambda}{\partial \theta}\right).$$

The Laplacian of the scalar field F, in Cartesian coordinates, is defined as

$$\nabla^2 F = \frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial y^2} + \frac{\partial^2 F}{\partial z^2},$$

which becomes

$$\nabla^2 F = \frac{1}{r^2\cos^2\theta}\frac{\partial^2 F}{\partial \lambda^2} + \frac{1}{r^2}\frac{\partial^2 F}{\partial \theta^2} + \frac{\partial^2 F}{\partial r^2} - \frac{\tan\theta}{r^2}\frac{\partial F}{\partial \theta} + \frac{2}{r}\frac{\partial F}{\partial r},$$

in spherical coordinates.

Finally, the spherical version of the Jacobian operator is given by

$$\frac{\partial\left(G_\lambda, G_\theta\right)}{\partial\left(\lambda, \theta\right)} = \frac{1}{a\cos\theta}\frac{\partial G_\lambda}{\partial \lambda}\left(\frac{1}{a}\frac{\partial G_\theta}{\partial \theta}\right) - \frac{1}{a}\frac{\partial G_\theta}{\partial \lambda}\left(\frac{1}{a\cos\theta}\frac{\partial G_\lambda}{\partial \lambda}\right).$$

A full explanation for all the operators can be found in [410].

12.3 Finite Differencing on the Sphere

In Chapters 8–10 we considered different discretizations in the horizontal plane, but we have to consider that we live on a near-spherical planet, and as such, for certain applications of geophysical modeling, we have to include the curvature of the Earth. In this section we explain different factors that can affect implementing finite difference schemes on the sphere.

12.3.1 Map Projections

There are many ways to represent data points on the sphere through different forms of canonical projections. In this section we shall briefly present some of the more commonly used projections for either numerical modeling on the sphere, or for the storing of observational data. There exist several desirable properties for any map projection; some of these are:

- Preservation of angles—This implies that the angles on the flat surface and the sphere should be equivalent.
- Preservation of area—Areas on the flat surface and the sphere should be identical.
- Preservation of shape—Shapes on the flat surface should match those on the sphere.
- Correct direction—Cardinal directions on the flat surface and the sphere should be equal.
- Shortest distance between two lines should be a great circle.

Unfortunately there does not exist an individual map projection that satisfies all of the five properties above. In atmospheric science applications, the most important requirement is that the angles be preserved. Map projections that preserve angles are referred to as **conformal** map projections. The most commonly used map projections are the polar stereographic, Mercator, and Lambert conic projections.

Stereographic projection

The stereographic projection is a mapping that projects the sphere on to a plane. This projection is defined on the whole sphere, except at the projection point. We first consider the unit sphere in Cartesian coordinates, which is given by $x^2 + y^2 + z^2 = 1$, and denote $N = (0, 0, 1)$, which is the coordinates for a north pole, where $\mathcal{M}$ here denotes the remainder of the sphere. If we now assume a plane that is $z = 0$ that runs through the center of the sphere, then the sphere's equator is the intersection of the sphere with the plane.

For any point P on $\mathcal{M}$, there is a unique line through N and P, that intersects the plane $z = 0$ at exactly one point, P'. We define the stereographic projection of P to be the point P' in the plane.

Given Cartesian coordinates (x, y, z) on the sphere, and (X, Y) on the plane; we can define the projection, and its inverse, by

$$(X, Y) = \left(\frac{x}{1 - z}, \frac{y}{1 - z} \right),$$
$$(x, y, z) = \left(\frac{2X}{1 + X^2 + Y^2}, \frac{2Y}{1 + X^2 + Y^2}, \frac{-1 + X^2 + Y^2}{1 + X^2 + Y^2} \right),$$

respectively. If we have the situation where we have spherical coordinate on the sphere denoted by (θ, λ) where $\theta \in [0, \pi]$, and $\lambda \in [0, 2\pi]$, and polar coordinates (R, Θ) on the plane, then the projections in these coordinates, and its inverse, are given by

$$(R, \Theta) = \left(\frac{\sin\theta}{1 - \cos\theta}, \lambda \right) = \left(\cot\frac{\theta}{2}, \lambda \right),$$
$$(\theta, \lambda) = \left(2\arctan\left(\frac{1}{R} \right), \lambda \right),$$

respectively, where it is assumed that $R = 0$ when $\theta = \pi$.

We present an example of the use of this projection in displaying model output in Fig. 12.2.

Mercator projection

The Mercator projection is a cylindrical map projection presented by the Flemish geographer and cartographer, Gerardus Mercator, in 1569. This map projection is practical for nautical applications due to its ability to represent lines of constant course, known as rhumb lines, as straight segments that conserve the angles with the meridians. Although the linear scale is equal in all directions around any point, thus preserving the angles and the shapes of small objects, the Mercator projection distorts the size of objects as the latitude increases from the equator to the poles, where the scale becomes infinite. A classic example of the distortion that this projection causes is that Greenland and Antarctica appear

FIGURE 12.2

Example of a stereographic projection.

much larger than they actually are relative to land masses near the equator, such as Central Africa. Another distortion this projection causes is that Greenland appears larger than Australia, while in actuality Australia is approximately three and a half times larger than Greenland. We have plotted an example of the Mercator projection to show the distortions in shapes that it causes in Fig. 12.3.

The following equations place the x-axis of the projection on the equator and the y-axis at longitude λ_0, such that

$$x = \lambda - \lambda_0,$$
$$y = \ln(\tan\theta + \sec\theta).$$

The formulas for the inverse of this projection are given by

$$\theta = 2\tan^{-1}\left(e^y\right) - \frac{1}{2}\pi,$$
$$\lambda = x + \lambda_0.$$

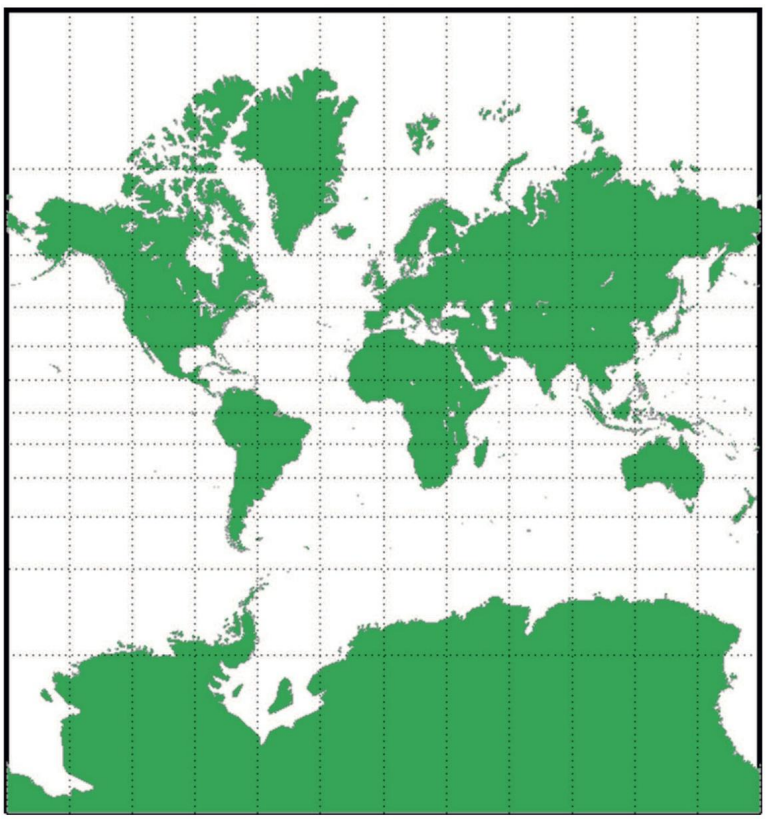


FIGURE 12.3

Example of a Mercator projection.

Lambert conic projection

A Lambert conformal conic projection is a conic map projection used for aeronautical charts, portions of the State Plane Coordinate System, and many national and regional mapping systems. The projection places a cone over the sphere of the Earth and projects the surface conformably onto the cone. The cone is then unrolled, and the parallel that was touching the sphere is assigned unit scale. That parallel is called the reference parallel or standard parallel.

By scaling the resulting map, two parallels can be assigned a unit scale, with scale decreasing between the two parallels and increasing outside them. As a result of scaling, deviation from unit scale can be minimized within a region of interest that lies largely between the two standard parallels.

Mathematically the projection and its inverse are defined as follows: let λ be the longitude, λ_0 be the reference longitude, θ be the latitude, where ϕ_0 is the reference latitude, and θ_1 and θ_2 are the standard parallels. Then the transformation of spherical coordinates to the plane via the Lambert conformal conic projection is given by

$$\begin{aligned} x &= \rho \sin\left(n\left(\lambda - \lambda_0\right)\right), \\ y &= \rho_0 - \rho\cos\left(n\left(\lambda - \lambda_0\right)\right), \end{aligned}$$

where

$$\begin{aligned} F &= \frac{\cos\theta_1 \tan^n\left(\frac{1}{4\pi} + \frac{1}{2\theta_1}\right)}{n}, \\ n &= \frac{\ln\left(\cos\theta_1 \sec\theta_2\right)}{\ln\left(\tan\left(\frac{1}{4\pi} + \frac{1}{2\theta_2}\right)\cot\left(\frac{1}{4\pi} + \frac{1}{2\theta_1}\right)\right)}, \\ \rho &= F\cot^n\left(\frac{1}{4\pi} + \frac{1}{2\theta}\right), \\ \rho_0 &= F\cot^n\left(\frac{1}{4\pi} + \frac{1}{2\theta_0}\right). \end{aligned}$$

The inverse formulas are

$$\begin{aligned} \theta &= 2\tan^{-1}\left(\left(\frac{F}{\rho}\right)^{\frac{1}{n}}\right) - \frac{1}{2\pi}, \\ \lambda &= \lambda_0 + \frac{\phi}{n}, \end{aligned}$$

where

$$\begin{aligned} \rho &= \operatorname{sign}(n)\sqrt{x^2 + (\rho_0 - y)^2}, \\ \phi &= \tan^{-1}\left(\frac{x}{\rho_0 - y}\right), \end{aligned}$$

while F, ρ_0, and n remain as defined above. An example of this projection can be seen in Fig. 12.4.

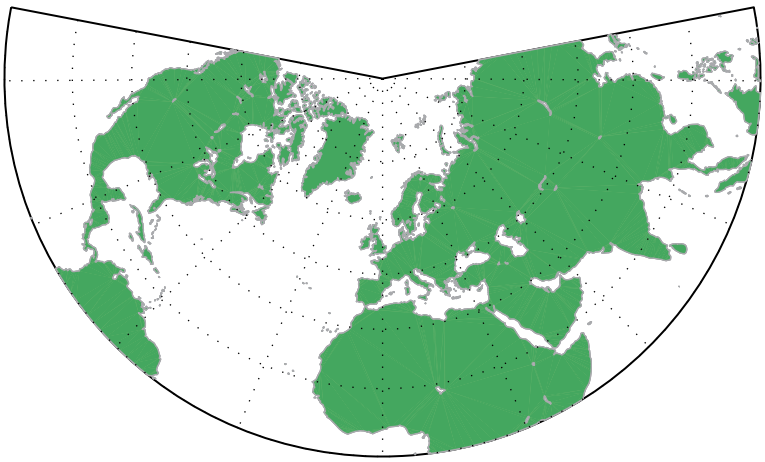


FIGURE 12.4

Example of a Lambert projection.

Sinusoidal projection

The sinusoidal projection is a pseudocylindrical equal area map projections; it is sometimes referred to as the Mercator equal area projection. The projection is defined by

$$x = (\lambda - \lambda_0)\cos\theta,$$
$$y = \theta,$$

where λ_0 is the central median. With this projection we have that scale is constant along the central median and the east-west scale is constant throughout the map. This then implies that the length of each parallel on the map is proportional to the cosine of the latitude. We have a plot of this projection in Fig. 12.5.

12.3.2 Grid-Point Representations of the Sphere

The idea of producing a weather forecast through the means of numerical approximations using physically based model was first proposed by Lewis Fry Richardson [366], who based his theory on Bjerknes's work, which had identified that numerical weather prediction was an initial value problem [41].

Richardson, recognizing that a weather forecast could be seen as an initial value problem, stated that if the values of certain environmental variables are known, then the physical equations can be used to calculate their value at some later time in the future. To achieve this, he proposed to divide the Earth's surface into a **grid**, where each grid cell was the base of a vertical column in the atmosphere. Each vertical column was then divided into several layers, which resulted in a three-dimensional grid of atmospheric boxes. The horizontal resolution that Richardson used was 3° in the longitudinal direction, and 200 km in the latitudinal direction, which then resulted in 12,000 columns to cover the globe. Given the location of each of these volumes, the vertical layers were at approximately 2, 4.2, 7.2, and 11 km

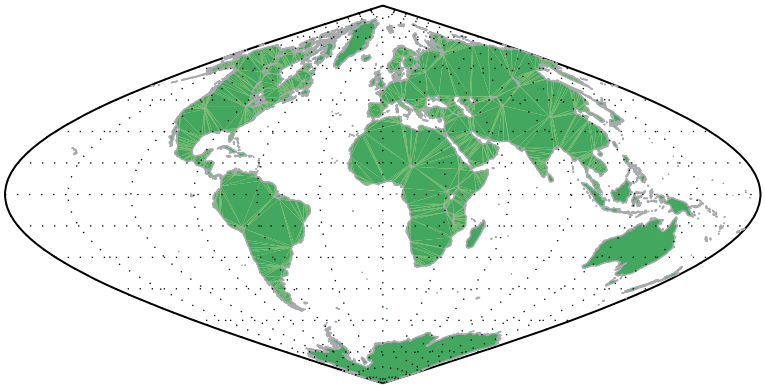


FIGURE 12.5

Example of a sinusoidal projection.

above the Earth's surface which is approximately equal to 800, 600, 400, and 200 hPa respectively. The values of the prognostic variables were stored at the center of each box. Unfortunately the initial forecast that Richardson was aiming for failed due to gravity waves that created noise in the observed data set, that then led to errors in the initial conditions to initialize the numerical model, a problem that can still plague numerical modeling today, though not as severely. We have presented a copy of the schematic of the original Richardson grid over northern Europe in Fig. 12.6.

As a result of the Richardson formulation of grid point approximations to the sphere, there has been much development of different grids that are comprised of different shapes, to be able to accommodate the spherical shape of the Earth, and to be able to represent the equations that govern the motion of the atmosphere, more accurately and more efficiently, given the computational power available. There has been development of grids that are based upon different shapes, where these shapes have been rectangular, triangular, and hexagonal, to name but a few. These types of spherical grid approximations are called *structured grids*.

Rectangular/square grids

If we consider the rectangular/square grid approximations, then this grid approximation has an equal number of points on each longitudinal line, implying that as we approaches the poles, the distance between the grid points decreases, which means that the resolution of the numerical approximation increases. However, this introduces what is referred to as the *pole/polar problem*, where the meridians converge to a point.

An alternative approach to the equal point grid formation was proposed by Kurihara in [229], who sought a formulation to keep the grid spacing uniform, yet still inside the latitude-longitude framework. The idea in [229] was to have a grid where the number of grid points along a latitudinal circle varied with the latitude. As a result of placing fewer points at the higher latitudes, then the grid covered the sphere more homogeneously. The formulation of the Kurihara grid starts with a single point at the pole and this latitude circle, which is a single point anyway, is given the label $j = 1$. The next latitude circle that is chosen is given the label $j = 2$, but there are $4\,(j-1)$ grid points along the

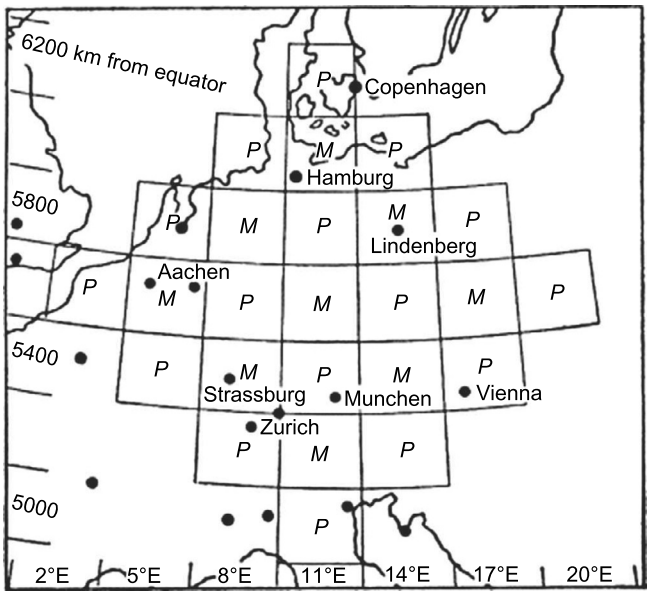


FIGURE 12.6

Copy of Figure 1 from [214] of the original Richardson grid for the first attempt at numerical weather prediction.

line of longitude of this circle. This carries on until we reach the equator where this circle is labeled $j = N + 1$, which represents the last of the **evenly** spaced points along the longitudinal meridian. The southern hemisphere is a mirror configuration of the northern hemisphere configuration just described. A copy of the illustration from Kurihara's grid configuration for an octant of the sphere is presented in Fig. 12.7.

The Kurihara grid configuration led to spuriously high pressure to develop at the poles, but it has been shown it is possible to remove this problem by using more accurate numerical schemes [353].

Triangular grids

It should be noted that triangular grids are not used as commonly as the rectangular grids in numerical models. One such form of a quasi-uniform grid whose base element is a triangle is referred to as the **spherical geodesic grid**. The idea of using triangles from an icosahedron was proposed in the 1960s by both Williamson [477] and Sadourny [373].

The spherical geodesic grid is constructed from an icosahedron that has 20 faces and 12 vertices. A simple scheme for constructing the geodesic grid is to divide the edges of the icosahedral triangular faces into smaller triangles, where the vertices of these triangles are the grid points. This process is recursively applied until the desired resolution is met. Each point on the face or edge of one of the faces of the icosahedron is surrounded by six triangles, making each grid point the center of a hexagon. The triangular faces of the icosahedrons are arranged into pairs to form five rhombuses around both poles.

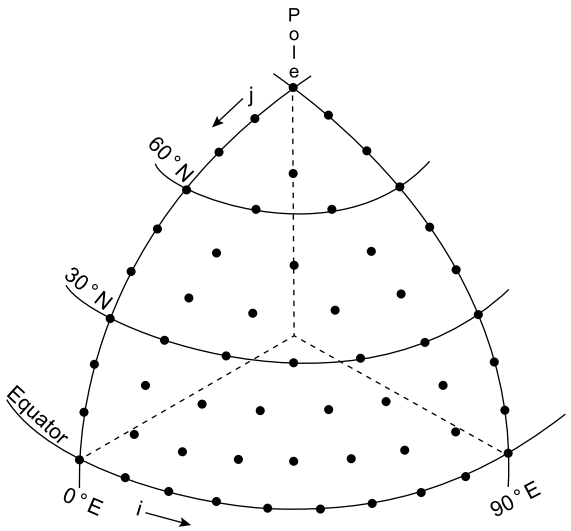


FIGURE 12.7

Copy of Figure 1 from [229] of the schematic of the proposed Kurihara grid.

Here the actual pole points are pentagonal points where the rhombuses meet. An example of this grid at different resolutions is presented in Fig. 12.8.

Hexagonal grid

We now present a summary of the hexagonal grid that is offered in [438]. The grid developed by Thuburn in [438] is similar to the twisted icosahedral grid from [171,172] but without the twisting. The sphere is covered by pentagonal and hexagonal grid boxes; the number of hexagons depended on the resolution, while there are always exactly 12 pentagons. Thuburn states that there is a hierarchy of possible grids that are of different resolutions, where each member of the hierarchy has approximately four times as many grid boxes as the previous member.

The starting point for the construction of the grid is the original regular icosahedron presented in the top left-hand plot in Fig. 12.8. Given a grid of triangles, a new finer grid of triangles is generated by placing new vertices at the midpoint of the existing edge and then projecting these new vertices onto the surface of the sphere (Fig. 12.8). This process is repeated, as we can see in Fig. 12.8, to obtain higher-resolution grids.

Before we progress any further in the description of this grid, we introduce the definition of the **Voronoi grid**.

Definition 12.1. Let X be a metric space with a distance function d. Let K be a set of indices and let P_k for $k \in K$ be an ordered collection of non-empty subsets in the space X. Then the Voronoi grid, or Voronoi cell, or Voronoi region, V_k, associated with the subset P_k, is the set of all points in X whose distance to P_k is not greater than their distance to the other subsets P_j, where j for $j \neq k$.

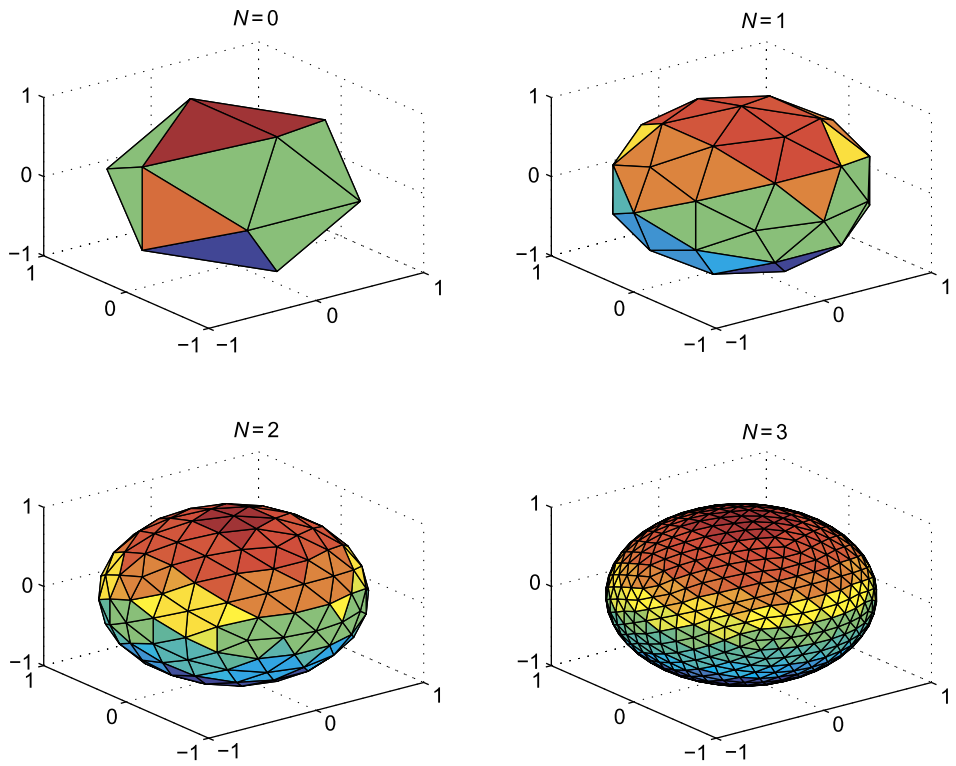


FIGURE 12.8

Plot of the build-up of the resolution of a triangular mesh.

Returning to the description from [438], then the hexagonal grid is a Voronoi grid associated with the triangular grid described above. Each vertex of the triangular grid corresponds to a face, that is to say, a grid box, of the Voronoi grid; each face of the triangular grid corresponds to a vertex of the Voronoi grid; and each edge of the triangular grid corresponds to an edge of the Voronoi grid. The edges of the Voronoi grid are the perpendicular bisectors of the edges of the triangular grid.

Note: The hexagonal grid boxes vary slightly in their exact shape and size. The most distorted hexagons are those immediately adjacent to the pentagons. However, the pentagons are perfectly regular.

A more mathematical description of the hexagonal grid can be found in [171], which we summarize here: given a set of N grid points $\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_N\}$ on the unit sphere S, then the definition of the Voronoi cell k associated with $\mathbf{P}_k$ is

$$\text{cell}_k = \{|\mathbf{p} - \mathbf{P}_k| \leq |\mathbf{p} - \mathbf{P}_l|\,,\ \forall \mathbf{p} \in S \text{ and } l \neq k\}\,, \tag{12.5}$$

where $|x_1 - x_2|$ is the distance function mention in the definition of the Voronoi grid, and this distance is a measure along the surface of the sphere. We now consider two neighboring grid points $\mathbf{P}_0$ and $\mathbf{P}_1$.

It is stated in [171] that the cell wall that is shared by these two points is the perpendicular bisector of the circular arc between the two points. A cell wall is a segment of a great circle. The end points of a wall are the Voronoi corners, the points where three cells intersect. The position of a corner is computed using three grid points. If we now consider the three points $\mathbf{P}_0$, $\mathbf{P}_1$, and $\mathbf{P}_2$ in three-dimensional Cartesian coordinates, $\mathbf{P}_0 = (x_0, y_0, z_0)$, $\mathbf{P}_1 = (x_1, y_1, z_1)$, and $\mathbf{P}_2 = (x_2, y_2, z_2)$, then the Voronoi corner, denoted by $\mathbf{C}$ in [171], is the point on the sphere that is equidistant from the three grid points, which is mathematically defined as

$$\mathbf{C} \equiv \frac{(\mathbf{P}_2 - \mathbf{P}_0) \times (\mathbf{P}_1 - \mathbf{P}_0)}{|(\mathbf{P}_2 - \mathbf{P}_0) \times (\mathbf{P}_1 - \mathbf{P}_0)|}. \tag{12.6}$$

In [171] it is noted that the points are ordered in a clockwise fashion to ensure that $\mathbf{C}$ lies in the same hemisphere as the three points. We have a copy of the build up of the resolution of the hexagonal grid from [171] in Fig. 12.9.

Cubed sphere

The **cubed sphere** is based on techniques similar to that of the geodesic grids, where now the initial shape is a cube. The derivation of the cubed sphere uses the same approach as that of the derivation

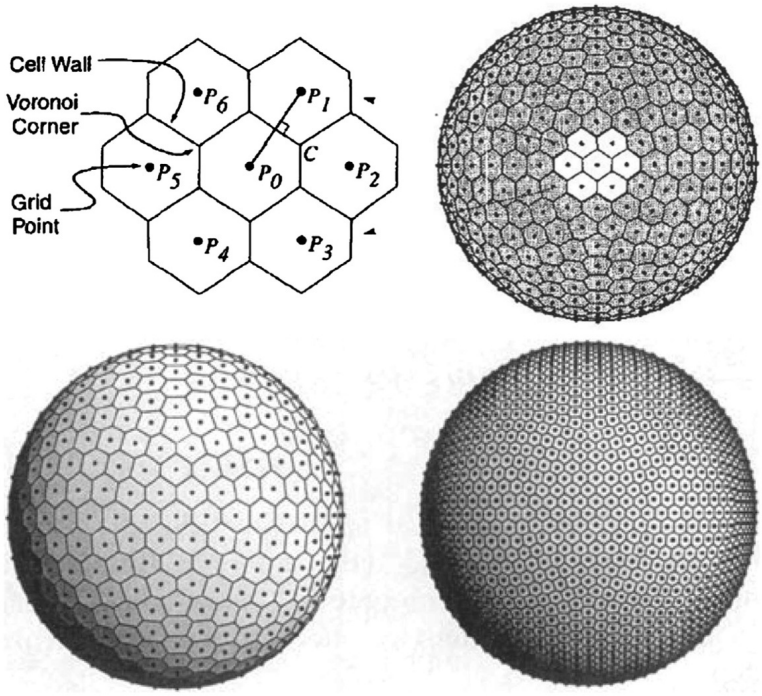


FIGURE 12.9

Copy of Figures 3 and 4 from [171] of the properties of the Voronoi grid and the twisted Icoshadren grid.

of the triangular and hexagonal grid, where we subdivide each face into smaller squares and then project on to the unit sphere. To show the progression of this subdividing arrives at the shape appearing approximately like a sphere, we have plotted the grids for $N = 2$, 4, 8, 16, 32, and 64 in Fig. 12.10, where N is the number of squares on each face of the cube in each direction. The cubed sphere was

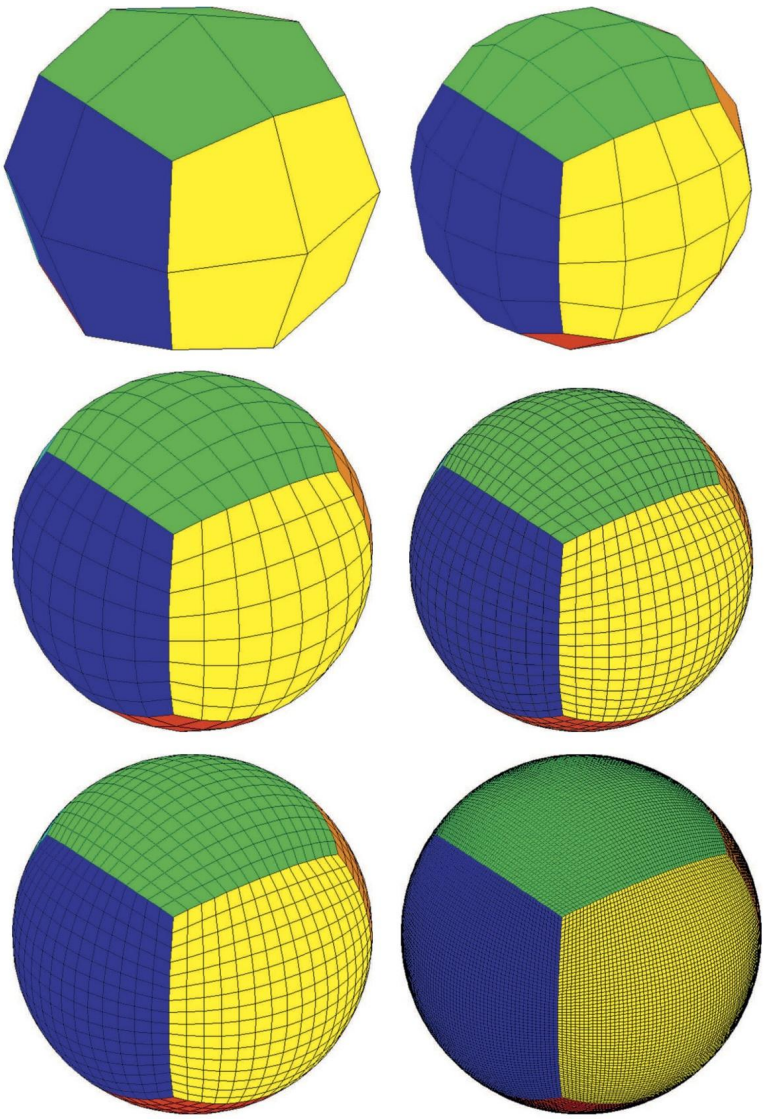


FIGURE 12.10

Plots of the build-up of the resolution of a cubed sphere grid.

presented in Sadourny in 1972 as a way of conserving finite differences approximations to the primitive equations [372]. It has also recently been used to model to a mantle convection simulation [203].

It is also part of the global weather prediction model at NASA, and since the first edition of this textbook it has been adapted as the grid for the global operational numerical weather prediction model at NOAA. The model at NOAA and NASA, where the numerics are included, but we have not make it that far, is referred to as the FV3 model which stands for Finite Volumes in 3-dimensions.

Yin-Yang

A recently derived approximation to the sphere was presented in Kageyama and Satoh [203] and Kageyama [204], where the motivation to developed this type of grid was not from a meteorological point of view, but for the simulation of geodynamo and mantle convection.

The Yin-Yang grid is referred to as an **overset grid**. The reason for the name is due to the grid appearing similar to the Chinese philosophy symbol of complementarity. The Yin-Yang grid is composed of two identical and complemental component grids.

The two component grids, just mentioned, are geometrically identical. The two component grids are referred to as the Yin grid, or n grid, and the Yang grid, or e grid. These two grids are combined to cover a spherical surface with partial overlap on their borders. The Yin-Yang grid is similar in appearance to the two patches that are stitched together to form a baseball. Each component grid is part of a latitude-longitude grid. The two grids overlap slightly and as such an interpolation is used to patch them together as two polar stereographic grids. We have presented copies of the Yin-Yang grid from [204] in Fig. 12.11.

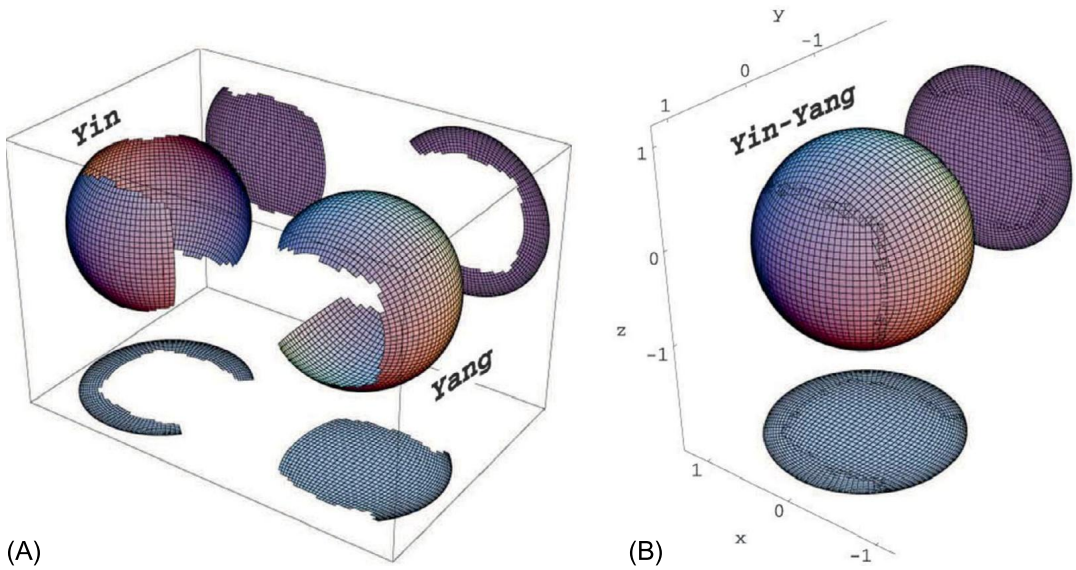


FIGURE 12.11

Copy of the Yin-Yang grid from [203].

Since the first edition of this textbook this grid has been adopted as the basis for the United Kingdom's Met. Office's operational global weather prediction system.

12.3.3 Different Grid Configuration

After the choice of the distribution of the grid points, i.e., rectangular, triangular, hexagonal, cubic sphere, Yin-Yang, etc., we now have to determine the arrangement of the prognostic variables on the grid. There are five different grids that are regularly used in numerical modeling which were presented in Arakawa and Lamb [12] for horizontal staggering. These five grid configurations are referred to as the Arakawa A, B, C, D, and E staggered grids. These grids are referred to as staggered due to the points where the height fields, h, and the wind components, u and v are stored at different locations. We shall present each of the five Arakawa grids in the following.

Arakawa A grid

The first of the Arakawa, not surprisingly, is referred to as the A grid, but ironically it is not a staggered grid. We have plotted the Arakawa A grid in Fig. 12.12.

The A-grid is referred to an *unstaggered* grid. An advantage of this grid is that it enables the Coriolis terms of the momentum equations to be easily evaluated, since the wind components are defined at the same points. While the A-grid may look like a good configuration for the grid points, this configuration can support two solutions that do not communicate with each other and that can be considerably different. As a result, the grid introduces noise at the smallest scales. Thus, this grid is not used in many applications nowadays.

Arakawa B grid

In this grid configuration we have the height fields, h stored at the corner of the grid cells, while the horizontal wind fields, (u, v), are stored at the center of the cells. We have plotted an illustration of this grid in Fig. 12.13.

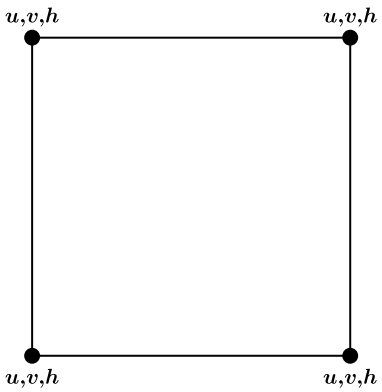


FIGURE 12.12

Diagram of the Arakawa unstaggered A grid.

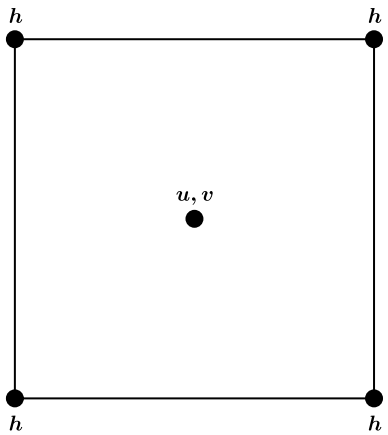


FIGURE 12.13

Diagram of the Arakawa staggered B grid.

Given this configuration for the B-grid; due to the wind fields being stored at the same points, evaluation of the Coriolis terms is made easier. However, the evaluation of the pressure gradient terms requires an averaging as is the case with the A-grid. The B-grid was used in the United Kingdom's Meteorological Office's older version of what they refer to as the unified model. They now use the C-grid which we introduce next.

Arakawa C grid

The C-grid is the first of two grids that does not have the wind fields collocated at the same grid point. In the C-grid configuration, the height fields are located at the center of the grid cell, while the u components of the wind fields are stored at the center of the vertices to the left and right of the center point, and the v components of the wind fields are stored at the center vertices above and below the center point. If we denote the center point as (i, j), where i represents the x-direction and j represents the y-direction, then the u fields are located at $\left(i \pm \frac{\Delta x}{2}, j\right)$ and the v fields are stored at $\left(i, j \pm \frac{\Delta y}{2}\right)$. We have plotted a schematic of this grid in Fig. 12.14.

An advantage of the C-grid is that it enables the gradients of the prognostic variables to be over Δx, compared to $2\Delta x$ for the A-grid. It is also used in the Weather, Research, and Forecasting (WRF) model, which is a community meso-scale numerical model available at the National Center for Atmospheric Research (NCAR), as well as being since the first edition, at the United Kingdom's Met. Office.

Arakawa D grid

The Arakawa D-grid is staged in the same way as the C-grid, but now the wind fields u and v are swapped. Therefore, we have that u fields are at $\left(i, j \pm \frac{\Delta y}{2}\right)$ and the v fields are stored at $\left(i \pm \frac{\Delta x}{2}, j\right)$. In [208] it is stated that there is no particular merit to this grid configuration. We have presented an illustration of the D-grid in Fig. 12.15.

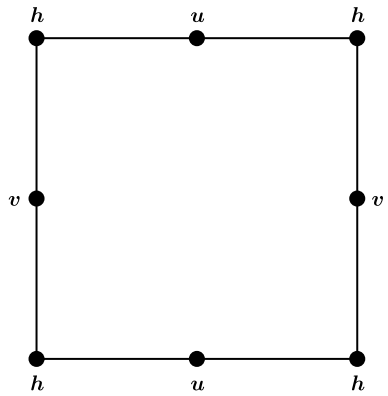


FIGURE 12.14

Diagram of the Arakawa staggered C grid.

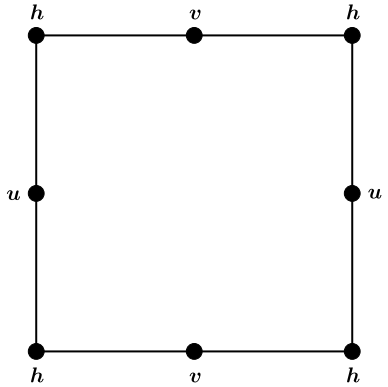


FIGURE 12.15

Diagram of the Arakawa staggered D grid.

Arakawa E grid

The last of the Arakawa grids is the E-grid, which is defined the same as the B-grid but the points where the wind fields are stored are rotated 45°. The E-grid can also be considered equivalent to superimposed, but shifted C-grids. We have plotted the schematic for the E-grid in Fig. 12.16.

12.3.4 Vertical Staggering Grids

When considering models that contain derivatives, as well as variables in the vertical, it becomes important to define the vertical coordinate system so that the numerical approximation is consistent with the dynamical situation. The most commonly used vertical coordinate systems are the z, pressure levels,

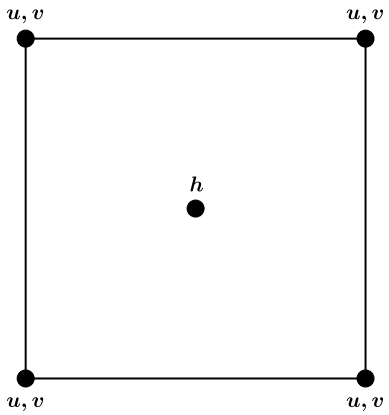


FIGURE 12.16

Diagram of the Arakawa staggered E grid.

p, a normalized pressure, σ, as presented in Phillips [336], potential temperature, θ [102], along with a combination of hybrid coordinate systems.

The normalized pressure coordinate system is referred to as the σ coordinates, where $\sigma \equiv \frac{p}{p_s}$, p_s is the pressure at the surface, and $\sigma = 1$ at the surface, and 0 at $p = 0$.

The grids that we have presented in the last subsection were associated with the staggering of the grid points in the horizontal direction. Most of the world's numerical weather prediction models of different scales have adopted a staggered grid in the vertical as well as in the horizontal. Two of the more regularly used vertical staggered grids are: (1) The Lorenz grid, which was presented in [269], where the vertical velocity is defined at the boundary of layers, while the prognostic variables are stored at the center of the layers. This grid configuration allows simple quadratic conservation, and the boundary conditions of no fluxes across the boundaries at the top and bottom are fulfilled. However, it was pointed out in [13] that this configuration allows the development of a spurious computational mode. (2) Another possible staggered vertical grid which has become used more frequently for numerical weather predictions models is the configuration presented in Charney and Phillips [60]. In this formulation we the height fields are at the σ levels, while the wind, and temperature, fields are stored at the halfway points between the sigma levels. This formulation is consistent with the hydrostatic equation. A schematic of these two vertical staggering approaches can be found in Fig. 12.17.

12.4 Introduction to Fourier Analysis

The first appearance of a very important technique in the solving of differential equations, which is referred to as the **separation of variables**, was in an 1807 paper titled ***Analytic Theory of Heat*** by the French mathematician Joseph Fourier. The initial version of this technique was the expression of a general function on the interval $[-\pi, \pi]$ in the form of the sum of a series of sines and cosines as

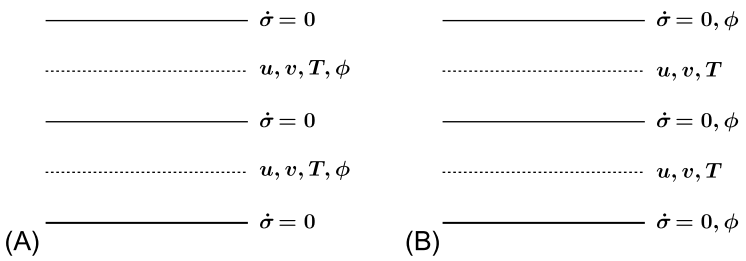


FIGURE 12.17

(A) The Lorenz staggered vertical grid, (B) The Charney-Phillips staggered vertical grid.

$$f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx), \tag{12.7}$$

where the coefficients a_n and b_n are **constants**. Fourier in the derivation of his method was able to find simple expressions for the coefficients just mentioned, and is the basic idea of **Fourier series**. It is also possible, through a simple change of variable, to alter the interval upon which we are solving from $[-\pi, \pi]$ to $[a, b]$. The notion of a Fourier series as defined in (12.7) can also be extended to a series of exponentials, of sines only or of cosines only. While it may not be obvious, the technique that Fourier introduced can be extended to express a given function in terms of other sets of *simpler functions*.

There is an analogous technique for functions defined over all of the real numbers $\mathbb{R}$, or on the interval $[0, \infty)$, instead of on an interval of a finite length. In this case it is possible to express the function, f, in the form

$$f(t) = \int_{-\infty}^{\infty} F(\omega) e^{it\omega} d\omega, \tag{12.8}$$

where the new function, F, can be determined. This is the basis of the idea of the **Fourier transforms**, which also provides a means for solving differential equations. The reason why the Fourier transform is valuable is that it enables us to convert the differentiation operation into a simpler one that involves the multiplication of independent variables.

Fourier series and transforms, have wide applicability in terms of solving differential equations, essentially by expressing a general function in terms of simpler ones. However, in the case of Fourier series and transforms we are dealing with sines, cosines of exponentials terms which occur naturally in various phenomena involving **waves**. This leads to a different aspect of Fourier techniques, where they can be viewed as a way of expressing a general signal in terms of its companion frequencies. That is to say that (12.8) can be thought of as showing how the general function f is made up of contributions from the oscillations of the form $e^{i\omega x}$ for all frequencies ω.

12.4.1 Fourier Series

Suppose that we wish to express a function $f : [-\pi, \pi] \to \mathbb{C}$ in the form

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty} (a_n \cos nx + b_n \sin nx). \tag{12.9}$$

It may not obvious how to proceed here, but we start by considering m and n as non-negative integers, along with the following integral:

$$\int_{-\pi}^{\pi} \cos mx \cos nx dx, \tag{12.10a}$$

$$= \int_{-\pi}^{\pi} \frac{1}{2} \left(\cos\left(m-n\right)x + \cos\left(m+n\right)x\right) dx,$$

$$= \left[\frac{1}{2}\frac{\sin\left(m-n\right)x}{m-n} + \frac{1}{2}\frac{\sin\left(m+n\right)x}{m+n}\right]_{-\pi}^{\pi}, \tag{12.10b}$$

where if $m \neq n$, then (12.10b) is equal to zero.

If we now consider the case where we have $m = n$, then the integral in (12.10a) becomes

$$\int_{-\pi}^{\pi} \left(\cos mx\right)^2 dx = \frac{1}{2}\int_{-\pi}^{\pi} \pi \left(1 + \cos\left(2mx\right)\right) dx, \tag{12.11}$$

which is equal to 2π when $m = 0$. For $m > 0$, then (12.11) is equal to π. Therefore, we have

$$\int_{-\pi}^{\pi} \cos mx \cos nx dx = \begin{cases} 0 & m \neq n \\ 2\pi & m = n = 0 \\ \pi & m = n > 0 \end{cases}.$$

Through a similar derivation to that above it is possible to show that

$$\int_{-\pi}^{\pi} \sin mx \sin nx dx = \begin{cases} 0 & m \neq n \\ \pi & m = n = 0 \\ 0 & m = n > 0 \end{cases},$$

and that

$$\int_{-\pi}^{\pi} \cos mx \sin nx dx = 0.$$

Now if we assume that we can express the function $f\left(x\right)$ as $f\left(x\right) = \frac{1}{2}a_0 + \sum_{n=1}^{N}\left(a_n \cos nx + b_n \sin nx\right)$, then we would have

$$\int_{-\pi}^{\pi} f\left(x\right)\cos mx dx = \int_{-\pi}^{\pi} a_0 \cos mx dx + \sum_{n=1}^{N}\int_{-\pi}^{\pi} a_n \cos nx \cos mx dx + \int_{-\pi}^{\pi} b_n \sin nx \cos mx dx, \tag{12.12}$$

$$= \begin{cases} \pi a_o & m = 0, \\ \pi a_m & m = 1, 2, \ldots, N, \\ 0 & m > N. \end{cases} \tag{12.13}$$

Through similar integration, it can easily be shown that

$$\int_{-\pi}^{\pi} \sin mx dx = \pi b_m \quad \text{for } m = 1, 2, \ldots, N. \tag{12.14}$$

If this works for a finite sum, then we may hope that it will work for a infinite sum. What these two expressions above tell us is that if the function f is well behaved and can be expressed in the form $\frac{1}{2}a_0 + \sum_{n=1}^{\infty}(a_n \cos nx + b_n \sin nx)$, then we expect

$$a_m = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos mx dx,$$
$$b_m = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin mx dx.$$

This enables us to arrive at the following definition.

Definition 12.2. Let $f : [-\pi, \pi] \to \mathbb{C}$ be an integrable function and define the **Fourier coefficients** of a_n, for $n = 0, 1, 2, \ldots$ and b_n for $n = 1, 2, \ldots$ as

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos nx dx, \tag{12.15a}$$
$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin nx dx, \tag{12.15b}$$

then the series defined by

$$\frac{1}{2}a_0 + \sum_{n=1}^{\infty}(a_n \cos nx + b_n \sin nx), \tag{12.16}$$

is the **Fourier series** of $f(x)$.

We now introduce an important definition that is needed for Fourier analysis as follows:

Definition 12.3. A sequence of functions, $e_n : [-\pi, \pi] \to \mathbb{C}$, is said to be **orthonormal** $\Leftrightarrow$

$$\int_{-\pi}^{\pi} e_m(x)\overline{e_n(x)}dx = \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases}.$$

As an example, if we define $e_0 = \frac{1}{\sqrt{2\pi}}$ and $\left.\begin{array}{l} e_{2n-1} = \frac{1}{\sqrt{\pi}}\sin nx \\ e_{2n} = \frac{1}{\sqrt{\pi}}\cos nx \end{array}\right\} n \in \mathbb{N}$, then applying the integral in Definition 12.3, results in

$$\int_{-\pi}^{\pi} e_m(x)\overline{e_n(x)}dx = \begin{cases} 1 & m = n \\ 0 & m \neq n \end{cases},$$

and as such this sequence is orthonormal. If we consider the sequence defined by $e_0 = \frac{1}{\sqrt{2\pi}}$, and $\left.\begin{array}{l} 2_{2n-1} = \frac{e^{inx}}{\sqrt{2\pi}} \\ e_{2n} = \frac{e^{-inx}}{\sqrt{2\pi}} \end{array}\right\} n \in \mathbb{Z}$, then this sequence is also orthonormal:

$$\int_{-\pi}^{\pi} e^{ipx}\overline{e^{iqx}}dx = \int_{-\pi}^{\pi} e^{i(p-q)x}dx = \frac{1}{i(p-x)}\left[e^{i(p-q)x}\right]_{-\pi}^{\pi} = 0,$$

for $p, z \in \mathbb{Z}$ and where $p - q \neq 0$.

Given the definition, and examples, of orthonormal sequences, we can now state, and prove, the following lemma.

Lemma 12.4. *Let $(e_n)_{n=1}^{\infty}$ be an orthonormal sequence of functions on the interval $[-\pi,\pi]$, and suppose that the function f such that $f:[-\pi,\pi]\to\mathbb{C}$ is integrable. Then among all the possible choices of $\lambda_1,\ldots,\lambda_n$ the one that minimizes the integral*

$$\int_{-\pi}^{\pi}\left|f(x)-\sum_{n=1}^{N}\lambda_n e_n(x)\right|^2 dx,$$

is given by

$$\lambda_n=\int_{-\pi}^{\pi}f(x)\overline{e_n(x)}dx. \tag{12.17}$$

Proof. Let $\alpha_n=\int_{-\pi}^{\pi}f(x)\overline{e_n(x)}dx$, then

$$\begin{aligned}
\int_{-\pi}^{\pi}\left|f(x)-\sum_{n=1}^{N}\lambda_n e_n(x)\right|^2 dx &= \int_{-\pi}^{\pi}\left|\left(f(x)-\sum_{n=1}^{N}\lambda_n e_n(x)\right)\overline{\left(f(x)-\sum_{m=1}^{N}\lambda_m e_m(x)\right)}\right|dx,\\
&=\int_{-\pi}^{\pi}\left|f(x)^2-\sum_{n=1}^{N}\lambda_n e_n(x)\overline{f(x)}-f(x)\sum_{m=1}^{N}\overline{\lambda_m e_m(x)}\right.\\
&\quad\left.+\sum_{n=1}^{N}\lambda_n e_n(x)\sum_{m=1}^{N}\overline{\lambda_m e_m(x)}\right|dx,\\
&=\int_{-\pi}^{\pi}|f(x)|^2dx-\sum_{n=1}^{N}\lambda_n\overline{\alpha_n}-\sum_{n=1}^{N}\overline{\lambda_n}\alpha_n+\sum_{m,n=1}^{N}\lambda_n\overline{\lambda_m}\int_{-\pi}^{\pi}e_n(x)\overline{e_m(x)}dx,\\
&=\int_{-\pi}^{\pi}|f(x)|^2dx-\sum_{n=1}^{N}\lambda_n\overline{\alpha_n}-\sum_{n=1}^{N}\overline{\lambda_n}\alpha_n+\sum_{m,n=1}^{N}\lambda_n\overline{\lambda_m},\\
&=\int_{-\pi}^{\pi}|f(x)|^2dx+\sum_{n=1}^{N}(\lambda_n-\alpha_n)\left(\overline{\lambda_n-\alpha_n}\right)-\sum_{n=1}^{N}|\alpha_n|^2,\\
&=\int_{-\pi}^{\pi}|f(x)|^2dx+\sum_{n=1}^{N}|\lambda_n-\alpha_n|^2-\sum_{n=1}^{n}|\alpha_n|^2.
\end{aligned}\tag{12.18}$$

The first and the last term in (12.18) are not dependent on λ_n and as such it is only the middle term which is, and the obvious choice for the λ_n to minimize the integral on the left above is to choose $\lambda_n=\alpha_n$ for $n=1,2,\ldots,N$, which is equivalent to (12.17), which completes the proof.

As with other theorems in this chapter, we shall present an example to illustrate this current lemma.

Example 12.5. *Let $e_0(x)=\frac{1}{\sqrt{2\pi}}$, $e_{2n}(x)=\frac{1}{\sqrt{\pi}}\cos nx$ and $e_{2n-1}(x)=\frac{1}{\sqrt{\pi}}\sin nx$, where $n\in\mathbb{N}$ and where the index begins at $n=0$, then*

$$\alpha_0 e_n(x)=\int_{-\pi}^{\pi}\left(f(x)\frac{1}{\sqrt{2\pi}}\right)dx\frac{1}{\sqrt{2\pi}}=\frac{1}{2\pi}\int_{-\pi}^{\pi}f(x)\,dx=\frac{1}{2}a_0, \tag{12.19}$$

where a_0 is the usual Fourier coefficient. Now if we consider the case for the even terms, we have

$$\alpha_{2n}e_{2n}(x) = \int_{-\pi}^{\pi} f(t)\frac{1}{\sqrt{\pi}}\cos ntdx\frac{1}{\sqrt{\pi}}\cos nx = a_n\cos nx. \tag{12.20}$$

Next, if we consider the odd terms, then we have

$$\alpha_{2n-1}e_{2n-1} = b_n\sin nx \text{ where } b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin nxdx. \tag{12.21}$$

Combining (12.19)–(12.21) results in

$$\sum_{n=1}^{2N}\alpha_n e_n(x) = \frac{1}{2}a_0 + \sum_{n=1}^{N}(a_n\cos nx + b_n\sin nx), \tag{12.22}$$

where (12.22) is the partial sum of the Fourier series of f. That is, by Lemma 12.4, the linear combinations of the functions $1, \cos x, \sin x, \cos 2x, \sin 2x, \ldots, \cos Nx, \sin Nx$, which is close to f in the series that it minimizes

$$\int_{-\pi}^{\pi}|f-g|^2dx,$$

where g is given by

$$g(x) = \frac{1}{2}a_0 + \sum_{n=1}^{N}(a_n\cos nx + b_n\sin nx). \tag{12.23}$$

Let $S_N(x) = \frac{1}{2}a_0 + \sum_{n=1}^{N}(a_n\cos nx + b_n\sin nx)$, then S_N can be thought of as a linear combination of $1, \cos x, \sin x, \ldots, \cos(N+1)x, \sin(N+1)x$, where the coefficients of the last two functions are zero, so because S_{N+1} is the closest such combination to f, we have that

$$\int_{-\pi}^{\pi}\left|f(x) - S_{N+1}(x)\right|^2dx \le \int_{-\pi}^{\pi}|f(x) - S_N(x)|^2dx. \tag{12.24}$$

We are now able to state the following corollary to bound the Fourier series coefficients:

Corollary 12.6. *Let $f:[-\pi,\pi]\to\mathbb{C}$ be integrable and let a_n and b_n be the Fourier coefficient, then $\sum_n|a_n|^2 + \sum_n|b_n|^2$ converges, and in particular $a_n\to 0$, $b_n\to 0$ as $n\to\infty$.*

The proof of this corollary is quite long and so is omitted here, but can be found in any good Fourier analysis textbook. However, there are a couple of definitions used in the proof of Corollary 12.6 that are important for the proofs of many of the Fourier analysis theorems that follow. Thus, if we define the sequence, c_n as $c_n \equiv \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)e^{-inx}dx$ for $n = 0, \pm1, \pm2, \ldots$, then

$$c_n + c_{-n} = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\left(e^{-inx} + e^{inx}\right)dx = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos nxdx = a_n, \tag{12.25a}$$

$$c_n - c_{-n} = \frac{1}{2\pi}\int_{-\pi}^{\pi} f(x)\left(e^{-inx} - e^{inx}\right)dx = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)(-i\sin nx)dx = -ib_n. \tag{12.25b}$$

This leads to the next main theorem associated with Fourier series approximations:

Theorem 12.7. *Let $F : [-\pi, \pi] \to \mathbb{C}$ be piecewise differentiable, and let $|x| < \pi$. Then if f is continuous at x, the Fourier series of f converges to $f(x)$.*

As an example, let $f(x) = x^2$ for $x \in [-\pi, \pi]$; this function is piecewise differentiable so the Fourier series holds:

$$\begin{aligned} a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} x^2 \cos nx\,dx &= \frac{1}{\pi}\left[x^2\frac{\sin nx}{n}\right]_{-\pi}^{\pi} - \frac{2}{n\pi}\int_{-\pi}^{\pi} x\sin nx\,dx, \\ &= \frac{2}{n\pi}\left[x\frac{\cos nx}{n}\right]_{-\pi}^{\pi} - \frac{2}{n^2\pi}\int_{-\pi}^{\pi}\cos nx\,dx, \\ &= \frac{2}{n\pi}\left(\frac{\pi(-1)^n}{n} - \frac{(-\pi)(-1)^n}{n}\right), \\ a_n &= \frac{4(-1)^n}{n^2}. \end{aligned}$$

The $n = 0$ term is calculated as

$$a_0 = \frac{1}{\pi}\int_{-\pi}^{\pi} x^2 dx = \frac{2\pi^2}{3}.$$

Finally, the b_n coefficients are calculated as

$$\begin{aligned} b_n &= \frac{1}{\pi}\int_{-\pi}^{\pi} x^2\sin nx\,dx, \\ &= \left[-\frac{1}{\pi}x^2\frac{\cos nx}{n}\right]_{-\pi}^{\pi} + \frac{2}{n\pi}\int_{-\pi}^{\pi} x\cos nx\,dx, \\ &= \frac{1}{\pi}\left(\frac{-\pi^2(-1)^n}{n} - \frac{(-\pi)^2(-1)^n}{n}\right) + \frac{2}{n^2\pi}[x\sin nx]_{-\pi}^{\pi} - \frac{2}{n^2\pi}\int_{-\pi}^{\pi}\sin nx\,dx, \\ b_n &= 0. \end{aligned}$$

Therefore, for $|x| < \pi$ we have that

$$x^2 = \frac{\pi^2}{3} + \sum_{n=1}^{\infty}\frac{4(-1)^n}{n^2}\cos nx.$$

Exercise 12.8. *Following the same derivation as presented for x^2, show that the Fourier series for $f(x) = x$ on $[-\pi, \pi]$ is*

$$x = \sum_{n=1}^{\infty}\frac{2(-1)^n}{n}\sin nx, \quad \textit{for } |x| < \pi.$$

If we plot the cosine function in the range of $x \in [-\pi, \pi]$, then we can see that it looks quite similar to a quadratic curve. Now, if we consider the sine function in the same interval, then we see that it approximates a straight line. An important feature of these functions is that they are periodic for values of x outside of $[-\pi, \pi]$. Given this information, it is possible to state a different version of Theorem 12.7 as follows:

Theorem 12.9. *Let* $f:[-\pi,\pi]\to\mathbb{C}$ *be piecewise differentiable and if* $|x|<\pi$, *then*

$$\frac{1}{2}a_0+\sum_{n=1}^{\infty}(a_n\cos nx+b_n\sin nx)=\frac{1}{2}\left(f\left(x^+\right)+f\left(x^-\right)\right), \tag{12.26}$$

where

$$a_n=\frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\cos ntdt,\quad n=0,1,2,\ldots,$$
$$b_n=\frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\sin ntdt,\quad n=1,2,\ldots.$$

Again to help illustrate this theorem, we consider the following example. Let $f(x)=\begin{cases}1 & 0\le x\le\pi\\ -1 & -\pi\le x\le 0\end{cases}$, then the Fourier coefficients for the Fourier series for this function are

$$\begin{aligned}
a_0&=\frac{1}{\pi}\int_{-\pi}^{\pi}f(x)\,dx=\frac{1}{\pi}\int_{-\pi}^{0}-1dx+\frac{1}{\pi}\int_{0}^{\pi}1dx=0,\\
a_n&=\frac{1}{\pi}\int_{-\pi}^{\pi}f(x)\cos nxdx=\frac{1}{\pi}\int_{-\pi}^{0}-\cos nxdx+\frac{1}{\pi}\int_{0}^{\pi}\cos nxdx=0,\\
b_n&=\frac{1}{\pi}\int_{-\pi}^{0}-\sin nxdx+\frac{1}{\pi}\int_{0}^{\pi}\sin nxdx,\\
&=\frac{1}{\pi}\int_{0}^{\pi}\sin(ny)\,dy+\frac{1}{\pi}\int_{0}^{\pi}\sin nxdx,\\
&=\frac{2}{\pi}\int_{0}^{\pi}\sin nxdx=\frac{2}{n\pi}\left[-\cos nx\right]_0^{\pi},\\
&=\frac{2}{n\pi}(1-\cos n\pi)=\begin{cases}0 & \text{n even}\\ \frac{4}{n\pi} & \text{n odd}\end{cases}.
\end{aligned}$$

Therefore, the Fourier series for $f(x)$ above is defined as $\sum_{n=1}^{\infty}\frac{4}{(2n-1)}\sin(2n-1)x=\begin{cases}1 & 0<x<\pi\\ 0 & x=0,\pm\pi\\ -1 & -\pi<x<0\end{cases}$.

This results in the important property relating odd and even functions to their Fourier series. If we assume that a function is odd $\forall x\in[-\pi,\pi]$, then $f_o(x)=-f_o(x)$ and so

$$b_n=\frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\sin ntdt=\frac{2}{\pi}\int_{0}^{\pi}f(t)\sin ntdt. \tag{12.27}$$

An even function is defined as $f_e(x)=\begin{cases}f(-x) & -\pi\le x\le 0\\ f(x) & 0<x\le\pi\end{cases}$. Because f_o is an odd function, all the cosine terms in its Fourier series have a zero coefficient, results in

$$\beta_n=\frac{2}{\pi}\int_{0}^{\pi}f(t)\sin ntdt=\frac{1}{\pi}\int_{-\pi}^{\pi}f_o(t)\sin ntdt, \tag{12.28}$$

which implies

$$\sum_{n=1}^{\infty} \beta_n \sin nx = \frac{1}{2}\left(f_o\left(x^+\right) + f\left(x^-\right)\right) = \frac{1}{2}\left(f\left(x^+\right) + f\left(x^-\right)\right), \tag{12.29}$$

provided $0 < x < \pi$.

Similarly, if the function f is even, then its Fourier series consists of just cosine terms, as

$$\frac{1}{2}\left(f_e\left(x^+\right) + f_e\left(x^-\right)\right) = \sum_{n=1}^{\infty} \alpha_n \cos nx, \tag{12.30}$$

where

$$\alpha_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos nx dx = \frac{2}{\pi}\int_{0}^{\pi} f(x)\cos nx dx. \tag{12.31}$$

For $0 < x < \pi$ we have $\frac{1}{2}\left(f_e\left(x^-\right) + f_e\left(x^+\right)\right) = \frac{1}{2}\left(f\left(x^+\right) + f\left(x^-\right)\right)$.

So far, the theory presented has been associated on the interval $[-\pi, \pi]$, but it could be the case that we need an approximation on the interval $[-a, a]$. A technique to address this is to start with the function $f : [-a, a] \to \mathbb{C}$, which is piecewise differentiable. Next we define a new function $\tilde{f} : [-\pi, \pi] \to \mathbb{C}$ by $\tilde{f}(x) = f\left(\frac{a}{\pi}x\right)$ for $-\pi \le x \le \pi$. This leads to

$$\frac{1}{2}\left(\tilde{f}\left(x^+\right) + \tilde{f}\left(x^-\right)\right) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\left(a_n \cos nx + b_n \sin nx\right), \tag{12.32}$$

where

$$\begin{aligned} a_n &= \frac{1}{\pi}\int_{-\pi}^{\pi} \tilde{f}(x)\cos nx dx, \\ &= \frac{1}{\pi}\int_{-\pi}^{\pi} f\left(\frac{a}{\pi}x\right)\cos nx dx, \\ &\equiv \frac{1}{a}\int_{-a}^{a} f(s)\cos\left(\frac{n\pi s}{a}\right) ds, \quad s \equiv \frac{ax}{\pi}, \\ b_n &= \frac{1}{a}\int_{-a}^{a} f(s)\sin\left(\frac{n\pi s}{a}\right) ds. \end{aligned}$$

Finally, let $y \in [-a, a]$ and $y = \frac{a}{\pi}x$, then we have

$$\begin{aligned} \frac{1}{2}\left(f\left(y^+\right) + f\left(y^-\right)\right) &= \frac{1}{2}\left(\tilde{f}\left(\frac{\pi}{a}y^+\right) + \tilde{f}\left(\frac{\pi}{a}y^-\right)\right), \\ &= \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\left(a_n \cos\left(\frac{n\pi y}{a}\right) + b_n \sin\left(\frac{n\pi y}{a}\right)\right). \end{aligned} \tag{12.33}$$

Again to help illustrate the results just presented, let us consider the following example:

Example 12.10. *A violin string is fixed at $x = 0$ and $x = a$, and stretched between these two points, displacement from its normal undisturbed position $y = 0$, is denoted by $y(x,t)$, where t represents time, and obeys the second-order partial differential equation*

$$\frac{\partial^2 y}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 y}{\partial t^2}, \quad \text{for } 0 \le x \le a, t \ge 0, \tag{12.34}$$

where $c = \sqrt{\frac{T}{p}}$, T is the tension of the string, and p is the mass per unit length. Given initial condition at $t = 0$ for the position of the string $y(x,0) = f(x)$, and the velocity of the string given by $\frac{\partial y}{\partial t}(x,0) = y(x)$ where the velocity and the position are both valid for $0 \le x \le a$, then find y.

We therefore require y to satisfy

$$\frac{\partial^2 y}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 y}{\partial t^2}, \quad \text{for } 0 \le x \le a, t \ge 0, \tag{12.35a}$$

$$y(0,t) = y(a,t) = 0, \quad \text{for } t \ge 0, \tag{12.35b}$$

$$\left.\begin{aligned} y(x,0) &= f(x) \\ \tfrac{\partial y}{\partial t}(x,0) &= g(x) \end{aligned}\right\}. \tag{12.35c}$$

Suppose that we seek a solution of (12.35a) and (12.35b) that is of the special form $y(x,t) = u(x)\,v(t)$, where u is a function of the spatial coordinate only, and v is a function of time only. We have that the trivial solution $y = 0$ is obviously a solution of this form. Thus we seek the **non-zero** solutions.

Given the form for the solution that we just assumed, from (12.35a) we have

$$\begin{aligned} u''v(t) &= \frac{1}{c^2}u(x)\,v''(t), \\ \Rightarrow \frac{u''(x)}{u(x)} &= \frac{1}{c^2}\frac{v''(t)}{v(t)}. \end{aligned} \tag{12.36}$$

Now the left-hand side of (12.36) depends on x only, and the right-hand side of (12.36) depends on t only, but they are equal, therefore, they both must be constant. We shall denote this constant by, λ, so that

$$\begin{aligned} u'' &= \lambda u, \\ v'' &= c^2\lambda v, \end{aligned}$$

also from (12.35b) we have $u(0)\,v^{(t)} = u(a)\,v(t) = 0$, $\forall t \ge 0$. Unless we have $v = 0$, which we have already said cannot be the solution, then $u(0) = u(a) = 0$.

Therefore, we have three possible situations that determine the solution to the spatial partial differential equation

$$u''(x) = \lambda u(x), \quad u(0) = u(a) = 0:$$

$\lambda < 0$, $\lambda = 0$, and $\lambda > 0$, and as such it can easily be shown that the three associated solutions are

$$u(x) = \begin{cases} A\cos\left(\sqrt{-\lambda}x\right) + B\sin\left(\sqrt{-\lambda}x\right) & (\lambda < 0) \\ A + Bx & (\lambda = 0) \\ A\cosh\left(\sqrt{\lambda}x\right) + B\sinh\left(\sqrt{\lambda}x\right) & (\lambda > 0) \end{cases}. \tag{12.37}$$

However, due to the boundary condition, $u(0) = 0$, the A constant for all three solutions in (12.37) is equal to zero. If we consider the boundary condition at $x = a$, which is $u(a) = 0$, then for the $\lambda = 0$ and $\lambda > 0$ cases we have that B is also equal to zero. If we now consider the case where $\lambda < 0$, then we have $u(a) = 0 \Rightarrow B \sin\left(\sqrt{-\lambda}a\right) = 0$, which implies that either $B = 0$ or $\sin\left(\sqrt{-\lambda}a\right) = 0$. As we are seeking the non-trivial solution, we must consider the case where $\sin\left(\sqrt{-\lambda}a\right) = 0$, which implies that if $a\sqrt{-\lambda} = n\pi$ for $n \in \mathbb{Z}$, then we have

$$-\lambda = \frac{n^2\pi^2}{a} \Rightarrow \lambda = -\frac{n^2\pi^2}{a^2},$$

for $n \in \mathbb{N}$ as $\lambda \neq 0$.

There are certain values of λ for which u can be non-zero,

$$u(x) = B \sin\left(\frac{n\pi}{a}x\right).$$

For the case where $\lambda = -\frac{n^2\pi^2}{a}$, we have

$$v''(t)\,\lambda i^2 v(t) = -\frac{n^2\pi^2c^2}{a^2}v(t), \tag{12.38}$$

so that

$$v(t) = C\cos\left(\frac{n\pi ct}{a}\right) + D\sin\left(\frac{n\pi ct}{a}\right), \tag{12.39}$$

for some constants C and D. Thus for each $n \in \mathbb{N}$ we have a solution of the form

$$\left(a_n \cos\left(\frac{n\pi ct}{a}\right) + b_n \sin\left(\frac{n\pi ct}{a}\right)\right)\sin\left(\frac{n\pi x}{a}\right). \tag{12.40}$$

We can add solutions of this form, and any finite sum of them, that will satisfy (12.35a) and (12.35b).

If the convergence is well enough behaved, then we can set

$$y(x,t) = \sum_{n=1}^{\infty}\left(a_n \cos\left(\frac{n\pi ct}{a}\right) + b_n \sin\left(\frac{n\pi ct}{a}\right)\right)\sin\left(\frac{n\pi x}{a}\right), \tag{12.41}$$

and y will still satisfy (12.35a) and (12.35b). Thus if we have $y(x,0) = \sum_{n=1}^{\infty} a_n \sin\left(\frac{n\pi x}{a}\right)$, and assuming we can differentiate term by term, then we obtain

$$\frac{\partial y}{\partial t}(x,0) = \sum_{n=1}^{\infty} b_n \frac{n\pi c}{a}\sin\left(\frac{n\pi x}{a}\right). \tag{12.42}$$

Thus, we need to choose a_n and b_n so that

$$\left.\begin{aligned}\textstyle\sum_{n=1}^{\infty} a_n \sin\left(\frac{n\pi x}{a}\right) &= f(x)\\ \textstyle\sum_{n=1}^{\infty} b_n \frac{n\pi c}{a}\sin\left(\frac{n\pi x}{a}\right) &= g(x)\end{aligned}\right\} \quad (0 \le x \le a).$$

Therefore, if we use a half-range series, then

$$a_n = \frac{2}{a}\int_0^a f(x)\sin\left(\frac{n\pi x}{a}\right)dx, \tag{12.43a}$$

$$b_n = \frac{2}{a}\int_0^a g(x)\sin\left(\frac{n\pi x}{a}\right)dx. \tag{12.43b}$$

Given the coefficients for a_n and b_n from (12.43a) and (12.43b), the solution

$$y(x,t) = \sum_{n=1}^{\infty}\left(a_n\cos\left(\frac{n\pi ct}{a}\right) + b_n\sin\left(\frac{n\pi ct}{a}\right)\right)\sin\left(\frac{n\pi x}{a}\right),$$

satisfies (12.35a)–(12.35c).

If we now consider a second example of the temperature $\theta(x,t)$ of a uniform insulated rod lying along the x-axis from 0 to a that satisfies the diffusion equation

$$\frac{\partial\theta}{\partial t} = \kappa\frac{\partial^2\theta}{\partial x^2} \quad (0 \le x \le a, t \ge 0), \tag{12.44}$$

and because the ends of the rod are insulated, we have the additional boundary conditions

$$\frac{\partial\theta}{\partial x}(0,t) = \frac{\partial\theta}{\partial x}(a,t) = 0 \quad (t \ge 0), \tag{12.45}$$

where the initial temperature distribution in the rod is given by

$$\theta(x,0) = f(x) \quad (0 \le x \le a), \tag{12.46}$$

then we need to find an expression for $\theta(x,t)$.

We again consider the solution to (12.44)–(12.46) to be of the form $\theta(x,t) = u(x)\,v(t)$; upon substituting this form of the solution into (12.44) we obtain

$$\kappa\frac{u''(x)}{u(x)} = \frac{v'(t)}{v(t)} = \lambda. \tag{12.47}$$

Noticing that left-hand side of (12.47) is independent of time, and the right-hand side is independent of the spatial coordinate again, implies that they must equal a constant λ as before. We therefore have the system of partial differential equations

$$u''(x) = \lambda u(x), \tag{12.48a}$$

$$v'(t) = \lambda\kappa v(t), \tag{12.48b}$$

that are subject to $u'(0) = u'(a) = 0$ from the boundary condition in (12.45). As we saw for the string example, the solutions to the differential equation $u'' - \lambda u$ are

$$u(x) = \begin{cases} A\cosh\left(\sqrt{\lambda}x\right) + B\sinh\left(\sqrt{\lambda}x\right) & (\lambda > 0) \\ A + Bx & (\lambda = 0) \\ A\cos\left(\sqrt{-\lambda}x\right) + B\sin\left(\sqrt{-\lambda}x\right) & (\lambda < 0) \end{cases}, \tag{12.49}$$

for constants A and B.

Unlike with the string example, we now have the boundary condition, $u'(0) = 0$, which implies that the constant B is equal to zero for all three solutions in (12.49). If we consider the remaining boundary condition $u'(a) = 0$, then for $\lambda > 0$ we have $A\sqrt{\lambda}\cosh\sqrt{\lambda}a = 0$, which implies that $A = 0$ for this case. For the $\lambda = 0$ solution this boundary condition provides no information; finally for the solution associate with $\lambda < 0$ we have $-\sqrt{-\lambda}A\sin\sqrt{-\lambda}a = 0$, and for non-zero A implies that $\sin\sqrt{-\lambda}a = 0$, which is to say that $\sqrt{-\lambda}a = n\pi$ for $n \in \mathbb{Z}$ and $n \neq 0$ hence $\lambda = -\frac{n^2\pi^2}{a^2}$ and

$$u(x) = A\cos\left(\frac{n\pi x}{a}\right) \quad (n \in \mathbb{N}).$$

Therefore, for $n = 0, 1, 2, \ldots$, the value of the constant is $\lambda = -\frac{n^2\pi^2}{a^2}$ and $v'(t) = \lambda\kappa v(t)$, which has the general solution

$$v(t) = Ce^{-\frac{n^2\pi^2\kappa}{a^2}t}. \tag{12.50}$$

Thus we shall choose our solution to be of the form

$$\theta(x,t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n\cos\left(\frac{n\pi x}{a}\right)e^{-\frac{n^2\pi^2\kappa}{a^2}t}, \tag{12.51}$$

and the solution in (12.51) satisfies (12.44) and (12.45). To satisfy (12.46) we require $\theta(x,0) = f(x)$, which implies

$$\frac{a_0}{2} + \sum_{n=1}^{\infty} a_n\cos\left(\frac{n\pi x}{a}\right) = f(x) \quad (0 \le x \le a). \tag{12.52}$$

We would then use a half-range cosine series, where the a_n Fourier coefficient are $a_n = \frac{2}{a}\int_0^a f(x)\cos\left(\frac{n\pi x}{a}\right)dx$.

As we mentioned during the string example, we assume that the convergence of the series is well behaved. We now state the theorem that defines when this is the case; where its proof can be found in any good Fourier analysis textbook.

Theorem 12.11. *Let $f : [-\pi, \pi] \to \mathbb{C}$ be a continuous and piecewise differentiable function and have a piecewise continuous derivative, and suppose that $f(\pi) = f(-\pi)$. Then the Fourier series of f converges uniformly.*

Example 12.12. *The displacement of a spring from its natural position is measured by $y(t)$, where t again denotes time. There is a restoring force from the spring; a damping force, and an external force $f(t)$ are applied. The value for y obeys the differential equation*

$$a\frac{d^2y}{dx^2} + b\frac{dy}{dx} + cy(t) = f(t), \tag{12.53}$$

where a, b, and c are all positive constants that are dependent of the characteristics of the situation. We shall assume that we have values for the initial conditions $y(0)$ and $y'(0)$, in which case it is possible to determine a solution, y, from (12.53).

From differential equation theory we know that the solution is the sum of a complementary function, y_{CF}, and a particular integral, y_{PI}, of the form

$$y = y_{CF} + y_{PI},$$

where y_{CF} is the solution to the homogeneous version of (12.53) and satisfies the two initial conditions as well. It can be shown that as a result of the three constants a, b, and c being positive, that $y_{CF} \to 0$ as $t \to \infty$, so that for a large value of time t, y is almost equal to y_{PI}.

We assume that the force applied to the spring is periodic with period $2T$ and is given by a saw-tooth function

$$f(t) = \begin{cases} t & (0 \le t \le T) \\ 2T - t & (T \le t \le 2T) \end{cases},$$

where for all value of t we have that $f(t + 2T) = f(t)$. To be able to solve a problem of this form, there are a few difficulties, but if the force were to be given by either $f(t) = \sin\left(\frac{n\pi t}{T}\right)$ or $f(t) = \cos\left(\frac{n\pi t}{T}\right)$ then it would be possible to apply the theory that has been presented with respect to Fourier series. In particular we can express the forcing f in terms of the Fourier series

$$f(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left(a_n \cos\left(\frac{n\pi t}{T}\right) + b_n \sin\left(\frac{n\pi t}{T}\right) \right). \tag{12.54}$$

Suppose that we can extend the force function to $[-T, 0]$ through periodicity, where we would have that $f(t) = -t$, which implies that the function is an even function, then we have already shown that $b_n = 0$ for all n. Therefore we have $a_n = \frac{2}{T} \int_0^T f(t) \cos\left(\frac{n\pi t}{T}\right) dt$, so that $a_0 = T$ and for all non-zero n, we have $a_n = 0$ for even values of n and $a_n = -\frac{4T}{n^2\pi^2}$ if n is odd. This implies that for $t \ge 0$, we have that the forcing can be expressed as

$$f(T) = \frac{T}{2} - \sum_{n=1}^{\infty} \frac{4T}{(2n-1)^2 \pi^2} \cos\left(\frac{(2n-1)\pi t}{T}\right). \tag{12.55}$$

In the general differential equation case, given by $ay'' + by'cy = e^{i\alpha t}$, it can be shown that this differential equation can be satisfied by a solution of the form

$$y(t) = \frac{e^{i\alpha t}}{-a\alpha^2 + bi\alpha + c}. \tag{12.56}$$

If we were to consider the real parts of the solution in (12.56) for the case where $f(t) = \cos\left(\frac{(2n-1)\pi t}{T}\right)$, then (12.53) would be satisfied by

$$y_n(t) = \frac{(c - a\alpha^2)\cos\left(\frac{(2n-1)\pi t}{T}\right) + b\alpha \sin\left(\frac{(2n-1)\pi t}{T}\right)}{(c - a\alpha^2)^2 + b^2\alpha^2}. \tag{12.57}$$

Thus a particular integral for (12.53), for the given forcing function $f(t)$, is given by

$$y(t) = \frac{T}{2b} - \sum_{n=1}^{\infty} \frac{4T}{(2n-1)^2 \pi^2} y_n(t). \tag{12.58}$$

This now leads to our final set of important theorems related to Fourier series that we shall not prove, but again can easily be found in any more detailed textbook on Fourier analysis.

Theorem 12.13. *Let $f:[-\pi,\pi]\to\mathbb{C}$ be piecewise continuous and let S_n denote the partial sum of the Fourier series of the function f, given by*

$$\frac{1}{2}a_0+\sum_{n=1}^{\infty}a_n\cos nx+b_n\sin nx,$$

then the partial sum S_n tends to the function f in what is referred to as in the mean in the sense that

$$\int_{-\pi}^{\pi}|f(t)-S_N(t)|^2\,dt\to 0 \text{ as } N\to\infty.$$

Theorem 12.14. *Let the functions f and g be such that $f,g:[-\pi,\pi]\to\mathbb{C}$ be piecewise continuous, then*

(i) $\frac{1}{2}|a_0|^2+\sum_{n=1}^{\infty}\left(|a_n|^2+|b_n|^2\right)=\frac{1}{\pi}\int_{-\pi}^{\pi}|f(t)|^2\,dt$; *and*
(ii) $\frac{1}{2}a_0\overline{\alpha_0}+\sum_{n=1}^{\infty}\left(a_n\overline{\alpha_n}+b_n\overline{\beta_n}\right)=\frac{1}{\pi}\int_{-\pi}^{\pi}f(t)\,\overline{g(t)}dt.$

Theorem 12.15. *Let $f:[-\pi,\pi]\to\mathbb{C}$ be a continuous and piecewise differentiable function, and let F be defined by*

$$F(x)=\int_0^x f(t)\,dt. \tag{12.59}$$

Then the function F is obtained by integrating the Fourier series for f term by term and is given by

$$F(x)=\frac{1}{2}a_0x+\sum_{n=1}^{\infty}\left(\frac{a_n}{n}\sin nx+\frac{b_n}{n}(1-\cos nx)\right). \tag{12.60}$$

12.4.2 Fourier Transforms

The main theorem about Fourier series, expressed in terms of exponentials rather than sines and cosine, is if $f:[-\pi,\pi]\to\mathbb{C}$ is well behaved, and if we set $c_n=\frac{1}{2\pi}\int_{-\pi}^{\pi}f(t)\,e^{-int}dt$, then we have $f(x)=\sum_{n=-\infty}^{\infty}c_ne^{inx}$.

If we change the variable $\frac{nx}{a}$, we see that on the interval $[-a,a]$, $f(x)=\sum_{n=-\infty}^{\infty}c_ne^{\frac{inx}{a}}$ where $c_n=\frac{1}{2a}\int_{-a}^{a}f(t)\,e^{\frac{in\pi t}{a}}dt$. This enables us to express the function, $f(x)$, as

$$f(x)=\sum_{n=-\infty}^{\infty}\frac{1}{2a}\int_{-a}^{a}f(t)\,e^{\frac{inx(t-x)}{a}}dt=\frac{1}{2\pi}\sum_{n=-\infty}^{\infty}\frac{\pi}{a}g\left(x,\frac{n\pi}{a}\right), \tag{12.61}$$

where

$$g(x,y)=\int_{-a}^{a}f(t)\,e^{iy(x-t)}dt. \tag{12.62}$$

We have evaluated $g(x, y)$ at intervals of $\frac{\pi}{a}$ and multiplied by the step length $\frac{\pi}{a}$ before summing, such that $\sum_{n=-\infty}^{\infty} g\left(x, \frac{n\pi}{a}\right)$ ought to be an approximation to $\int_{-\infty}^{\infty} g(x, y)\, dy$ as $a \to \infty$, and as such the step length $\frac{\pi}{a} \to 0$. Therefore, we might expect the sum to tend to the integral, which results in

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} g(x, y)\, dy = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ixy} \left(\int_{-\infty}^{\infty} f(t)\, e^{-ity} dt \right) dy. \tag{12.63}$$

This does turn out to be true. The important thing to notice is that if we calculate $\int_{-\infty}^{\infty} f(t)\, e^{-ity} dt$, then, by carrying out the second interval, we can recover f. This then enables us to state the following definition:

Definition 12.16. Suppose that $f : \mathbb{R} \to \mathbb{C}$ and that $\forall y \in \mathbb{R}$, then the improper integral

$$\int_{-\infty}^{\infty} f(t)\, e^{-ity} dt, \tag{12.64}$$

exists. Thus we say that the function, $F : \mathbb{R} \to \mathbb{C}$, defined by

$$F(y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)\, e^{-ity} dt, \tag{12.65}$$

is the **Fourier transform** of f.

A features to notice here is that if the improper integral $\int_{-\infty}^{\infty} |f|\, dt$ exists and if f is piecewise continuous, then the integral $\int_{-\infty}^{\infty} f(t)\, e^{-ity} dt$ will exist, and as such we have

$$\left| \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} f(t)\, e^{-ity} dt \right| \leq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} |f(t)| \left| e^{-ity} \right| dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} |f(t)|\, dt. \tag{12.66}$$

We now consider the following example: let $f(t) = \begin{cases} 1 & |t| \leq 1 \\ 0 & |t| > 0 \end{cases}$, then we have

$$\begin{aligned} F(y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)\, e^{-ity} dt = \frac{1}{\sqrt{2\pi}} \int_{-1}^{1} e^{-ity} dt, \\ &= \frac{1}{\sqrt{2\pi}} \frac{e^{-iy} - e^{iy}}{-iy} = \sqrt{\frac{2}{\pi}} \frac{\sin y}{y} \quad (y \neq 0). \end{aligned}$$

For the case of $y = 0$, we have $F(0) = \sqrt{\frac{2}{\pi}}$.

We now consider

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(y)\, e^{ixy} dy = \frac{1}{\sqrt{2\pi}} \sqrt{\frac{2}{\pi}} \int_{-\infty}^{\infty} e^{ixy} \frac{\sin y}{y} dy = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{\sin y}{y} e^{ixy} dy. \tag{12.67}$$

For the case where $a \neq 0$, we have

$$\int_1^b \frac{e^{iay}}{y} dy = \left[\frac{e^{iay}}{iay} \right]_1^b + \int_1^b \frac{e^{iay}}{iay^2} dy = \frac{e^{iab}}{iab} - \frac{e^{ia}}{ia} + \int_1^b \frac{e^{iay}}{iay^2} dy.$$

Since the integral $\int_1^\infty \frac{1}{y^2} dy$ exists, the right-hand side above tends to a limit as $b \to \infty$ provided that $a \neq 0$. Similarly the integral $\int_{-a}^1 \frac{e^{iay}}{y} dy$ exists. Therefore, we have

$$\int_{-\infty}^{\infty} \frac{\sin}{y} e^{ixy} dy = \int_{-\infty}^{\infty} \frac{1}{2i} \frac{e^{-(x+1)y} - e^{i(x-1)y}}{y} dy, \tag{12.68}$$

and this integral exists if $x \pm 1 \neq 0$. Thus we have

$$\begin{aligned}
\frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ixy} dy \int_{-\infty}^{\infty} f(t) e^{-ity} dt &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ixy} 2 \frac{\sin y}{y} dy, \\
&= \lim_{b \to \infty} \frac{1}{\pi} \int_{-b}^{b} e^{ixy} \frac{\sin y}{y} dy, \\
&= \lim_{b \to \infty} \int_0^b \frac{\sin y}{y} \left(e^{-ixy} + e^{ixy} \right) dy, \\
&= \lim_{b \to \infty} \frac{2}{\pi} \int_0^b \frac{\sin y \cos xy}{y} dy, \\
&= \lim_{b \to \infty} \frac{1}{\pi} \int_0^b \frac{\sin (x+1) y}{y} - \frac{\sin (x-1) y}{y} dy, \\
&= \frac{1}{\pi} \int_0^\infty \frac{\sin (x+1) y}{y} dy - \frac{1}{\pi} \int_0^\infty \frac{\sin (x-1) y}{y} dy, \\
&= \phi(x+1) - \phi(x-1),
\end{aligned}$$

where

$$\phi(x) = \frac{1}{\pi} \int_0^\infty \frac{\sin (xy)}{y} dy.$$

By changing the variable to $u = xy$, we see that for $x > 0$, $\phi(x) = \phi(1)$. For $x < 0$ we have $\phi(x) = -\phi(1)$. This then gives us

$$\phi(x+1) - \phi(x-1) = \begin{cases} 0 & x > 1 \\ 2\phi(1) & -1 < x < 1 \\ 0 & x < -1 \end{cases}. \tag{12.69}$$

Since $\phi(1) = \frac{1}{2}$, we have that $\phi(x+1) - \phi(x-1) = f(x)$ for $x \neq \pm 1$.

Given the results from the example above, the inverse Fourier transform is defined as follows:

Definition 12.17. If $f : \mathbb{R} \to \mathbb{C}$ has a Fourier transform, then its **inverse Fourier transform** is defined to be the function whose value at y is

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{ity} dt. \tag{12.70}$$

An important feature to note here is that $F(-y) = f$; that is to say that the value of the inverse Fourier transform of f is $F(-y)$.

This leads to the following two important lemmas:

Lemma 12.18. ***Riemann-Lebesgue Lemma:***

(i) Suppose that $f:[a,b]\to\mathbb{C}$ *is piecewise continuous, then*

$$\int_a^b f(t)\,f(t)\,e^{-ity}dt\to 0 \text{ as } y\to\pm\infty. \tag{12.71}$$

(ii) Suppose that $f:\mathbb{R}\to\mathbb{C}$ *is piecewise continuous and that* $\int_{-\infty}^{\infty}|f|$ *exists, then*

$$\int_{-\infty}^{\infty} f(t)\,e^{-ity}dt\to 0 \text{ as } y\to\pm\infty. \tag{12.72}$$

Lemma 12.19. *Suppose that the function* $g:\mathbb{R}\to\mathbb{C}$ *is piecewise continuous and that* $\int_{-\infty}^{\infty}|g(t)\,t|\,dt$ *exists; if*

$$G(y)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} g(t)\,e^{-ity}dt \tag{12.73}$$

is the Fourier transform of g, then G is differentiable and is given by

$$G'(y)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}-itg(t)\,e^{-ity}dt. \tag{12.74}$$

Given these two lemmas, it is possible to state the important Fourier integral theorem as:

Theorem 12.20. ***Fourier Integral Theorem:*** *Let* $f:\mathbb{R}\to\mathbb{C}$ *be piecewise differentiable and suppose that the integral,* $\int_{-\infty}^{\infty}|f|$ *exists, and if* f *is continuous at* x*, then*

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}e^{ity}\left(\int_{-\infty}^{\infty}f(t)\,e^{-ity}dt\right)dy=f(x), \tag{12.75}$$

that is to say that

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}F(y)\,e^{ixy}dy=f(x).$$

We shall prove neither of these lemmas nor the Fourier integral theorem, but we do consider the following example to help illustrate this theory.

Suppose that we have the function $f(x)=e^{|x|}$ which is continuous and piecewise differentiable, and that $\int_{-\infty}^{\infty}e^{-|x|}dx=2$. We have presented a plot of the function $e^{-|x|}$ in Fig. 12.18.

Then let F be the Fourier transform of f given by

$$\begin{aligned}
F(y)&=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}f(t)\,e^{-ity}dt,\\
&=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}e^{-|x|}e^{-ity}dt,\\
&=\frac{1}{\sqrt{2\pi}}\int_{0}^{\infty}e^{-t-ity}dt,\\
&=\frac{1}{\sqrt{2\pi}}\int_{0}^{\infty}e^{-t(1+iy)}+e^{-t(1-iy)}dt,
\end{aligned}$$

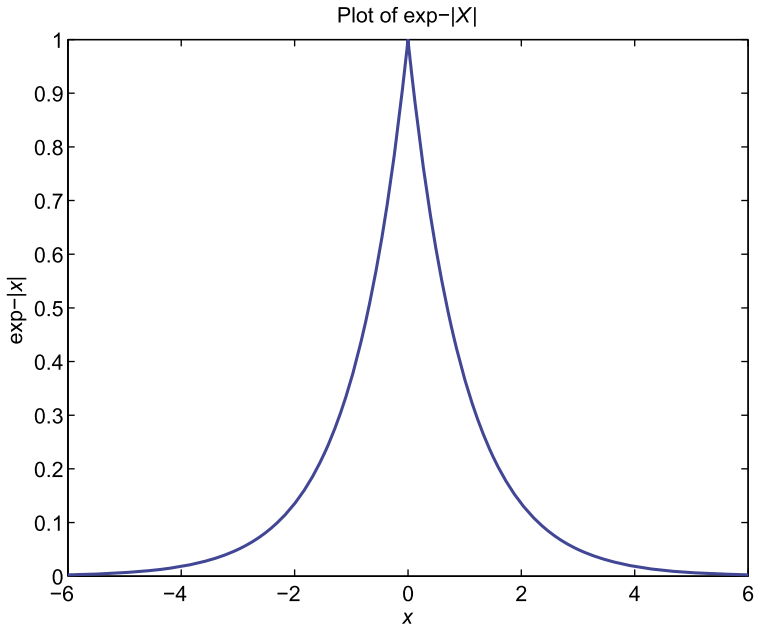


FIGURE 12.18

Plot of the exponential of the negative absolute value.

$$= \frac{1}{\sqrt{2\pi}}\left[-\frac{e^{-t(1+iy)}}{(1+iy)} - \frac{e^{-t(1-iy)}}{(1-iy)}\right]_{t=0}^{t\to\infty},$$

$$= \frac{1}{\sqrt{2\pi}}\left(\frac{1}{1+iy} + \frac{1}{1-iy}\right) = \frac{1}{\sqrt{2\pi}}\frac{2}{1+y^2} = \sqrt{\frac{2}{\pi}}\frac{1}{1+y^2} = F(y). \tag{12.76}$$

By Fourier's integral theorem, we have that

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} F(y)\, e^{ixy} dy = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{e^{ixy}}{1+y^2} dy = f(x) = e^{-|x|}. \tag{12.77}$$

We now consider some properties of the Fourier transforms: let $\hat{f}$ be the Fourier transform of the function f, then we have

$$\hat{f}(y) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} f(t)\, e^{-ity} dt, \tag{12.78a}$$

$$\widehat{f+g} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} (f(t)+g(t))\, e^{-ity} dt = \hat{f}(y) + \hat{g}(y), \tag{12.78b}$$

$$\widehat{(\lambda f)} = \lambda \hat{f}. \tag{12.78c}$$

Now suppose that $\int_{-\infty}^{\infty} |f|$ and $\int_{-\infty}^{\infty} |y'|$ both exist and that $f(x) \to 0$ as $x \to \pm\infty$, then the Fourier transform of f', denoted as $\widehat{f'}$, is given by

$$\begin{aligned}
\widehat{f'}(y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f'(t)\, e^{-ity} dt, \\
&= \underbrace{\frac{1}{\sqrt{2\pi}} \left[f(t)\, e^{-ity} \right]_{t=-\infty}^{t=\infty}}_{=0} - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)\, (-iy)\, e^{-ity} dt, \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} iy f(t)\, e^{-ity} dt, \\
&= (iy)\, \hat{f}(y). \qquad (12.79)
\end{aligned}$$

The significance of the result in (12.79) is that it implies that the Fourier transform applied to a derivative converts the process of differentiation into a simple multiplication process.

To help illustrate the result in (12.79), recall the temperature of the rod example, temperature is denoted as $\theta(x, t)$ at ordinate x and time t in a uniform insulated rod lying along the x-axis that obeys the diffusion equation

$$\frac{\partial \theta}{\partial t} = \kappa \frac{\partial^2 \theta}{\partial x^2} \quad (x \in \mathbb{R}, t \geq 0),$$

where κ is a positive constant. At the initial time, $t = 0$ the distribution of the temperature throughout the rod is given $\theta(x, 0) = f(x)$ for $x \in \mathbb{R}$. Therefore we need to find θ.

The first thing to note is that the relevant set of values of x is $\mathbb{R}$, not a finite interval, therefore we shall use a Fourier transform for the temperature, denoted by Θ, and is defined as

$$\Theta(y, t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \theta(x, t)\, e^{-ixy} dx. \qquad (12.80)$$

Substituting (12.80) into the diffusion equation above, results in

$$\begin{aligned}
\frac{\partial \Theta}{dt} &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \frac{\partial \theta}{\partial t}(x, t)\, e^{-ixy} dx, \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \kappa \frac{\partial^2 \theta}{\partial x^2}(x, t)\, e^{-ixy} dx, \\
&= \kappa (iy)^2 \Theta(y, t),
\end{aligned}$$

which comes about through the property derived in (12.79), extended to the second derivative. We should note here that this is only possible under the assumption that the functions are well enough behaved so that we can differentiate under the integral sign.

Therefore we have

$$\frac{\partial \Theta}{\partial t} = -\kappa y^2 \Theta(y, t), \qquad (12.81)$$

where for fixed y, (12.81) is an ordinary differential equation for t, thus

$$\Theta(y,t) = A(y)\, e^{-\kappa y^2 t},$$

where $A(y)$ depends on y only. Therefore,

$$A(y) = \Theta(y,0) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \theta(x,0)\, e^{-ixy} dx = \hat{f}(y), \tag{12.82}$$

where $\hat{f}$ is the Fourier transform of f. Thus,

$$\Theta(y,t) = \hat{f}(y)\, e^{-\kappa y^2 t}.$$

By using the inverse Fourier transform, we have

$$\theta(x,t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \Theta(y,t)\, e^{ixy} dy = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(y)\, e^{-\kappa y^2 t} e^{ixy} dy.$$

We now introduce another property of the Fourier transforms which is associated with the convolution of functions.

Definition 12.21. Suppose that the two functions f and g are such that $f, g : \mathbb{R} \to \mathbb{C}$, and are piecewise continuous and that the two integrals $\int_{-\infty}^{\infty} |f|$ and $\int_{-\infty}^{\infty} |g|$ both exist, then we define the convolution of the functions f and g, denoted by $f * g$, as

$$(f * g)(x) = \int_{-\infty}^{\infty} f(x-y)\, g(y)\, dy. \tag{12.83}$$

Through the change of variable $z = x - y$, we see that

$$(f * g)(x) = \int_{-\infty}^{\infty} f(z)\, g(x-z)\, dx = (g * f)(x). \tag{12.84}$$

Given the definition for the convolution of functions above, we can now state an important theorem about the Fourier transform of the convolution of functions.

Theorem 12.22. *Suppose that the two functions f and g are such that $f, g : \mathbb{R} \to \mathbb{C}$, and are piecewise continuous and that $\int_{-\infty}^{\infty} |f|$ and $\int_{-\infty}^{\infty} |g|$ both exist, then*

$$\widehat{(f * g)}(y) = \sqrt{2\pi}\, \hat{f}(y)\, \hat{g}(y). \tag{12.85}$$

As with many of the other theorems presented in this chapter, we shall omit the proof, but again it can be found in most textbooks on Fourier analysis.

If we consider the function $f(x) = e^{-ax^2}$ for $a > 0$ which is important for our infinite heat rod example, then $\hat{f}(y)$ is given by

$$\begin{aligned} \hat{f}(y) &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-ax^2 - ixy} dx, \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-a\left(x + \frac{i}{2a}y\right)^2 - \frac{y^2}{4a}} dx, \end{aligned}$$

$$= e^{-\frac{y^2}{4a}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-a\left(x+\frac{i}{2a}y\right)^2} dx.$$

If we now introduce the change of variable $s = x + \frac{i}{2a}y$, then this implies

$$\hat{f}(y) = e^{-\frac{y^2}{4a^2}} \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-as^2} ds. \tag{12.86}$$

The step above is justified by a field of mathematics called *complex analysis* which is the area of mathematics that deals with integration and differentiation involving complex numbers. We shall not go into details here of how those techniques are applied to the situation in hand, but move on, assuming that those properties hold.

We now introduce the change of variable $t = as^2$, implying that $s = \sqrt{\frac{t}{a}}$, that makes the integrand $ds = \frac{1}{2}\frac{1}{\sqrt{at}}dt$. Given this information, we can rewrite the integral in (12.86) as

$$\begin{aligned} 2\int_0^{\infty} e^{-as^2} ds &= 2\int_0^{\infty} e^{-t} \frac{1}{2\sqrt{at}} dt, \\ &= \frac{1}{\sqrt{a}} \int_0^{\infty} t^{-\frac{1}{2}} e^{-t} dt. \end{aligned} \tag{12.87}$$

To progress further, we need to identify the integral in (12.87) which we have seen before in an earlier chapter. Surprisingly the integral in (12.87) is a gamma function, specifically it is $\Gamma\left(\frac{1}{2}\right)$, where from earlier has the value $\sqrt{\pi}$. Therefore, we have

$$\hat{f}(y) = \frac{1}{\sqrt{2a}} e^{-\frac{y^2}{4a}}. \tag{12.88}$$

Returning to the heat flow problem in an infinite rod, let $g(x) \equiv e^{-\frac{x^2}{4\kappa t}}$, and

$$\frac{1}{\sqrt{\frac{1}{2\kappa t}}} e^{-\frac{y^2}{\kappa t}} = \sqrt{2\kappa t} e^{-\frac{y^2}{\kappa t}}.$$

This implies that our Fourier-transformed solution Θ can be expressed as

$$\Theta(y,t) = \hat{f}(y) e^{-\kappa y^2 t} \equiv \underbrace{\left(\sqrt{2\pi}\,\hat{f}(y)\,\hat{g}(y)\right)}_{\widehat{(f*g)}} \frac{1}{\sqrt{2\kappa t}\sqrt{2\pi}}. \tag{12.89}$$

Thus the expression for Θ in (12.89) is $\sqrt{2\pi}$ times the product of two Fourier transforms and as such θ must be the convolution of two functions:

$$\begin{aligned} \theta(x,t) &= \frac{1}{\sqrt{4\kappa t\pi}} (f * g), \\ &= \frac{1}{\sqrt{4\kappa t\pi}} \int_{-\infty}^{\infty} f(s)\, g(x-s)\, ds, \end{aligned}$$

$$= \frac{1}{\sqrt{4\kappa t\pi}} \int_{-\infty}^{\infty} f(s)\, e^{-\frac{(x-s)^2}{4\kappa t}}\, ds.$$

We now move on to consider the situation similar to how we defined the half-range Fourier series by extending a function on $[0, a]$ to $[-a, a]$, either as an odd or an even, function, but here we consider how to define two new Fourier transforms on the interval $[0, \infty]$.

Definition 12.23. Let $f : [0, \infty) \to \mathbb{C}$ be a piecewise differentiable function and suppose that $\int_0^\infty |f|$ exists. Then we define the **Fourier sine transform**, $\hat{f}_s$, and the **Fourier cosine transform**, $\hat{f}_c$, as

$$\hat{f}_s(y) = \sqrt{\frac{2}{\pi}} \int_0^\infty f(y) \sin(xy)\, dx, \tag{12.90a}$$

$$\hat{f}_s(y) = \sqrt{\frac{2}{\pi}} \int_0^\infty f(y) \cos(xy)\, dx. \tag{12.90b}$$

An important feature to notice is that $\hat{f}_s(-y) = -\hat{f}_s(y)$ and $\hat{f}_c(-y) = \hat{f}_c(y)$ and so we have all of the information in $\hat{f}_s$ and $\hat{f}_c$ if we consider $y \geq 0$. This then leads to the following theorem:

Theorem 12.24. *Let $f : [0, \infty) \to \mathbb{C}$ be piecewise differentiable and suppose that $\int_0^\infty |f|$ exists. Then if $x > 0$,*

$$\begin{aligned} \frac{1}{2}\left(f\left(x^+\right) + f\left(x^-\right)\right) &= \sqrt{\frac{2}{\pi}} \int_0^\infty f(y) \cos(xy)\, dy, \\ &= \sqrt{\frac{2}{\pi}} \int_0^\infty f(y) \sin(xy)\, dy. \end{aligned}$$

If $x = 0$, then the right-hand side of the equation above has a value zero for the sine transform and $f\left(0^+\right)$ in the cosine case.

Again we shall omit the proof of this theorem here, but present an example to help illustrate how it works. Let $f(x) = e^{-x}$, then the Fourier cosine transform of this function is given by

$$\begin{aligned} \hat{f}_c(y) &= \sqrt{\frac{2}{\pi}} \int_0^\infty e^{-x} \cos(xy)\, dx = \sqrt{\frac{2}{\pi}} Re\left(\int_0^\infty e^{-x} e^{ixy} dx\right) = \sqrt{\frac{2}{\pi}} Re\left(\int_0^\infty e^{-x(1-iy)} dx\right), \\ &= \sqrt{\frac{2}{\pi}} Re\left(\frac{1}{1-iy}\right) = \sqrt{\frac{2}{\pi}} \frac{1}{1+y^2}. \end{aligned}$$

By taking imaginary parts, we have

$$\hat{f}_s(y) = \sqrt{\frac{2}{\pi}} Im\left(\int_0^\infty e^{-x(1-iy)} dx\right) = \sqrt{\frac{2}{\pi}} \frac{y}{1+iy^2}.$$

Using the inversion rule, we have that

$$e^{-x} = \sqrt{\frac{2}{\pi}} \int_0^\infty \hat{f}_c(y) \cos(xy)\, dy = \frac{2}{\pi} \int_0^\infty \frac{\cos(xy)}{1+y^2} dy, \quad \text{for } x \geq 0.$$

Similarly,

$$e^{-x} = \sqrt{\frac{2}{\pi}} \int_0^\infty \hat{f}_s(y) \sin(xy)\, dy = \frac{2}{\pi} \int_0^\infty \frac{y \sin(xy)}{1+y^2} dy, \quad \text{for } x > 0.$$

This leads to the following lemma linking the sine and cosine Fourier transforms:

Lemma 12.25. *Suppose that the function $f : [0, \infty) \to \mathbb{C}$ is differentiable, along with f' being piecewise continuous, and that $f(x) \to 0$ and $x \to \infty$ and finally that both f and f' have Fourier sine and cosine transforms, then*

$$\widehat{f'}_s(y) = -y \hat{f}_c(y),$$
$$\widehat{f'}_c(y) = -\sqrt{\frac{2}{\pi}} f(0) + y \hat{f}_s(y).$$

An important feature to note about Lemma 12.25 is that to evaluate $\widehat{f'}_s$, we **do not** need to know $f(0)$, but to evaluate $\widehat{f'}_c$ we **do** require $f(0)$.

Example 12.26. *The temperature $\theta(x,t)$ at ordinate x and time t in a uniform insulated rod of heat-conducting material lying along the **positive** x-axis obeys the diffusion equation*

$$\frac{\partial \theta}{\partial t} = \kappa \frac{\partial^2 \theta}{\partial x^2} \quad (x, t > 0),$$

where κ is a positive constant and the initial temperature distribution throughout the rod is given by

$$\theta(x, 0) = f(x) \quad (x \geq 0).$$

If the end of the rod at $x = 0$ is maintained at a constant temperature a, find θ.

Given the description of the problem above, we can see that we are solving this problem over the quarter plane.

We cannot apply the cosine or sine Fourier transforms on the time component as the function is constant, since $\theta(0, t) = a$, and we have that there does not exist a Fourier sine or cosine transform of a non-zero constant. Therefore, we shall either take the Fourier sine or cosine transform with respect to x, that is, transforming x into another variable.

Since we know $\theta(0, t)$ but we do not have information about $\frac{\partial \theta}{\partial x}(0, t)$, we choose to use the Fourier sine transform, because if we were to use the Fourier cosine transform we would need to know $\frac{\partial \theta}{\partial x}(0, t)$.

Therefore, let

$$\Theta(y, t) = \sqrt{\frac{2}{\pi}} \int_0^\infty \theta(x, t) \sin(xy)\, dx,$$

then

$$\begin{aligned}
\frac{\partial \Theta}{\partial t}(y, t) &= \sqrt{\frac{2}{\pi}} \int_0^\infty \frac{\partial \theta}{\partial t}(x, t) \sin(xy)\, dx, \\
&= \kappa \sqrt{\frac{2}{\pi}} \int_0^\infty \frac{\partial^2 \theta}{\partial x^2}(x, t) \sin(xy)\, dx, \\
&= -\kappa y \sqrt{\frac{2}{\pi}} \int_0^\infty \frac{\partial \theta}{\partial x}(x, t) \cos(xy)\, dx,
\end{aligned}$$

$$= -\kappa y \left(\sqrt{\frac{2}{\pi}} \theta(0,t) + \sqrt{\frac{2}{\pi}} \int_0^\infty \theta(x,t) \sin(xy)\, dx \right),$$
$$= \kappa y \sqrt{\frac{2}{\pi}} a - \kappa y^2 \Theta(y,t),$$

where we have used both properties for the Fourier sine and cosine transforms from Lemma 12.25 in the derivation above.

We now have that for fixed y, the following ordinary differential equation

$$\frac{\Theta(y,t)}{\partial t}(y,t) = -\kappa y^2 \Theta(y,t) + \kappa y \sqrt{\frac{2}{\pi}} a,$$

has to be solved. This leads to

$$\Theta(y,t) = A(y) e^{-\kappa y^2 t} + \sqrt{\frac{2}{\pi}} \frac{a}{y},$$

where $A(y)$ is independent of t. By using the initial conditions, we have that

$$\Theta(y,0) = \sqrt{\frac{2}{\pi}} \int_0^\infty \theta(x,0) \sin(x,y)\, dx,$$
$$= \hat{f}_s(y),$$

and hence we find

$$A(y) = \hat{f}_s(y) - \sqrt{\frac{2}{\pi}} \frac{a}{y}.$$

This enables us to find an expression for $\Theta(y,t)$ as

$$\Theta(y,t) = \left(\hat{f}_s(y) - \sqrt{\frac{2}{\pi}} \frac{a}{y} \right) e^{-\kappa y^2 t} + \sqrt{\frac{2}{\pi}} \frac{a}{y},$$
$$= \hat{f}_s(y) e^{-\kappa y^2 t} + \sqrt{\frac{2}{\pi}} \frac{a}{y} \left(1 - {}^{-\kappa y^2 t} \right).$$

Therefore, we can now express the temperature, $\theta(x,t)$ as

$$\theta(x,t) = \sqrt{\frac{2}{\pi}} \int_0^\infty \Theta(y,t) \sin(x,y)\, dy,$$
$$= \sqrt{\frac{2}{\pi}} \int_0^\infty \left(\hat{f}_s(y) e^{-\kappa y^2 t} + \sqrt{\frac{2}{\pi}} \frac{a}{y} \left(1 - {}^{-\kappa y^2 t} \right) \right) \sin(xy)\, dy.$$

Another view of Fourier transforms

There are many phenomena that involve oscillations of some form: for example, in sound, light, and electrical circuits. In some phenomena it is simple to find the response of the system concerned to an input which is a pure sine wave, that is, an input of the form $A\cos(\omega t)$, where A and ω are constants, and t represents time. A here is the amplitude of the wave that completes a period in $\frac{2\pi}{\omega}$ seconds so that

its frequency is $\frac{\omega}{2\pi}$ cycles per second. A wave like this also has a phase that needs not take its maximum value at $t = 0$, and we introduce another constant, α, to account for this, to obtain $A\cos(\omega t + \alpha)$.

As an example, if a voltage, denoted by V, which is given by $A\cos(\omega t)$ is applied to an electrical circuit that has a resistance R and an inductance, L, then the resulting current is

$$\frac{A}{\sqrt{R^2 + \omega^2 L^2}} \cos(\omega t - \alpha), \tag{12.91}$$

where $\alpha = \tan^{-1} \frac{\omega L}{R}$. In this case the response of the circuit to an input is dependent on the frequency, ω.

Surprisingly it is actually easier to deal with this problem by introducing complex numbers. We notice straightaway that $A\cos(\omega t) = Re\left(Ae^{i\omega t}\right)$ and that

$$\frac{A}{\sqrt{R^2 + \omega^2 L^2}} \cos(\omega t - \alpha) = Re\left(H(\omega) e^{i\omega t}\right), \tag{12.92}$$

where $H(\omega) = \frac{e^{-i\alpha}}{\sqrt{(R^2+\omega^2 L^2)}}$. That is to say that is possible to express the response of the system by multiplying the input by a single complex number $H(\omega)$.

To be able to use the property that we have just observed, we need to decompose an input signal, $f(t)$, into components of the form $e^{i\omega t}$, which is what the Fourier transform accomplishes.

If we assume that the input signal is described by $f(t)$, then we have that

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(\omega) e^{i\omega t} d\omega, \tag{12.93}$$

which indicates how much of each simple frequency $e^{i\omega t}$ the signal $f(t)$ contains. The next step is to assume that the process is linear, so that the output corresponding to the sum of several inputs is the sum of the outputs corresponding to the separate inputs, $e^{i\omega t}$, changed to $H(\omega) e^{i\omega t}$ for each ω and the output added to obtain

$$g(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{f}(\omega) H(\omega) e^{i\omega t} d\omega. \tag{12.94}$$

An interesting case of this is a **filter**, which is a device that suppresses certain frequencies.

Sampling and recovering signals

Suppose that $f(t)$ represents some signal. In many cases it is known that only frequencies in a certain range occur in f. In the case of sound waves we know that there is only a certain range of frequencies that the human ear can hear, and as such there would be little point producing frequencies about that if the end result were to be listened to. As an example, the signal produced by a telephone can be assumed to contain only the frequencies between certain limits, leaving the other frequencies available for other purposes. Suppose also that f is real-valued, so that its Fourier transform satisfies

$$\hat{f}(-\omega) = \overline{\hat{f}(\omega)}. \tag{12.95}$$

If f has frequencies that only occur in a certain band, then $\hat{f}(\omega) = 0$ for $|\omega| > b$. Thus it is possible to expand the Fourier transform $\hat{f}$ in a Fourier series on $[-b, b]$ as,

$$\hat{f}(\omega) = \sum_{n=-\infty}^{\infty} c_n e^{i\frac{n\pi\omega}{b}} \quad (\omega \in [-b, b]), \tag{12.96}$$

where

$$c_n = \frac{1}{2b}\int_{-b}^{b} \hat{f}(\omega)\, e^{i\frac{n\pi\omega}{b}} d\omega. \tag{12.97}$$

Since we already know that the sum of the Fourier series is periodic, and $\hat{f}(\omega) = 0$ unless $\omega \in [-b, b]$, we see that

$$\hat{f}(\omega) = \chi(\omega) \sum_{n-\infty}^{\infty} c_n e^{i\frac{n\pi\omega}{b}}, \tag{12.98}$$

where $\chi(\omega) = 1$ if $\omega \in [-b, b]$, and $\chi(\omega) = 0$ otherwise.

However, because $\hat{f}$ is a Fourier transform, and is zero outside $[-b, b]$, we have

$$c_n = \frac{1}{2b}\int_{-\infty}^{\infty} \hat{f}(\omega)\, e^{i\frac{n\pi\omega}{b}} d\omega = \frac{\sqrt{2\pi}}{2b} f\left(-\frac{n\pi}{b}\right), \tag{12.99}$$

by the Fourier Integral theorem. Therefore, the coefficients, c_n, are values of the function f that are sampled at intervals $\frac{\pi}{b}$.

Therefore,

$$\begin{aligned} f(t) &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \hat{f}(\omega)\, e^{i\omega t}, \\ &= \sum_{n=-\infty}^{\infty} \frac{c_n}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \chi(\omega)\, e^{i\frac{n\pi\omega}{b} + i\omega intt} d\omega, \\ &= \sum_{n=-\infty}^{\infty} f\left(-\frac{n\pi}{b}\right) \frac{\sin\left(\left(\frac{n\pi}{b} + t\right)b\right)}{n\pi + tb}. \end{aligned} \tag{12.100}$$

For

$$\begin{aligned} \int_{-\infty}^{\infty} \chi(\omega)\, e^{i\frac{n\pi\omega}{b} + i\omega t} d\omega &= \int_{-b}^{b} e^{i\omega\left(\frac{n\pi}{b} + t\right)} d\omega, \\ &= \frac{2\sin\left(\frac{n\pi}{b} + t\right)b}{\frac{n\pi}{b} + t}. \end{aligned}$$

If we now let $T = \frac{\pi}{b}$ be the interval between successive points at which f is sampled, and substitute $m = -n$ into (12.100), then we obtain

$$\begin{aligned} f(t) &= \sum_{m=-\infty}^{\infty} f(mT) \frac{\sin(tb - m\pi)}{b(t - mT)}, \\ &= \sum_{m=-\infty}^{\infty} f(mT) \frac{(-1)\sin(tb)}{b(t - mT)}. \end{aligned}$$

An important feature that we need to note here is that in practice any real-life signal, f, will only last for a finite time, which then means that $f(mT)$ will be non-zero for only a finite number of values of m. Suppose that the signal lasts for T_0 seconds, which represents the time from $t = 0$ to $t = T$, and that the frequencies involved are all less than B Herz. Given that the frequency is $\frac{\omega}{2\pi}$, so that $\frac{\omega}{2\pi} \leq B$, we choose $b = 2\pi B$. The non-zero terms will be from $m = 0$ to $m = \frac{T_0}{T} = 2T_0 B$.

In the example that we have presented here we needed to sample f, which means that we had to evaluate it at intervals of $T = \frac{B}{2}$, where B is the highest frequency in the signal in Herz. For us to carry this out we need to know how wide the band is, because the wider the band, the closer together the sampling points need to be.

There is another interesting characteristic of oscillations. If we let

$$f(t) = \begin{cases} 1 & t \in [0, a] \\ 0 & \text{elsewhere} \end{cases},$$

then we have

$$\hat{f}(\omega) \sqrt{\frac{2}{\pi}} e^{-\frac{1}{2}ia\omega} \frac{\sin\left(\frac{1}{2}a\omega\right)}{\omega}.$$

Therefore, as a increase, $\frac{2\pi}{a}$ decreases so that the width of the graph of $\hat{f}$ decreases as that of the original function f increases. Thus we require some way of measuring the spread of a function about its center. This can be achieved through the following theorem:

Theorem 12.27. *Suppose that the functions f and g are such that $f, g : \mathbb{R} \to \mathbb{C}$ and are continuous, bounded, and that $\int_{-\infty}^{\infty} |f|$ and $\int_{-\infty}^{\infty} |g|$ exist. Then*

$$\int_{-\infty}^{\infty} f(t)\, \overline{g(t)} dt = \int_{-\infty}^{\infty} \hat{f}(\omega)\, \overline{\hat{g}(\omega)} d\omega, \tag{12.101}$$

where in particular

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} \left|\hat{f}(\omega)\right|^2 d\omega. \tag{12.102}$$

12.4.3 Laplace Transforms

We have seen how the Fourier transform can be used to solve certain differential equations. This usefulness arises because once we pass to the transform of a function, certain operations are transformed into simpler operations, especially differentiation operators being transformed into multiplication operators.

The Fourier transform has certain drawbacks and not every function has a Fourier transform; for example, constant functions do not have a Fourier transform. To avoid this deficiency, and to deal with the set of functions that occur in ordinary differential equations, we introduce another transform known as the **Laplace transform**.

Definition 12.28. Suppose that the function, f, is such that $f : [0, \infty) \to \mathbb{C}$, is piecewise continuous, and that there is a real number α such that $f(t) e^{-\alpha t}$ is a bounded function of $t \in [0, \infty)$. Then we define the **Laplace transform** of f, denoted as $Lf : (\alpha, \infty) \to \mathbb{C}$, as

$$(Lf)(p) = \int_0^{\infty} f(t) e^{-pt} dt. \tag{12.103}$$

Remark 12.29. If $f(t)e^{-\alpha t}$ is bounded, where here we mean that $\forall t \in [0,\infty)$ we have that $\left|f(t)e^{-\alpha t}\right| \leq M$ for some M, then if $p > \alpha$ we have

$$\left|f(t)e^{-pt}\right| \leq \left|f(t)e^{-\alpha t}e^{-(p-\alpha)t}\right| < Me^{-(p-\alpha)t},$$

and since $\int_0^\infty e^{-(p-\alpha)t}dt$ exists, we see that the Laplace transform of f exists for $p > \alpha$.

Examples:

(i) Let $f(t) = 1$, then its associated Laplace transform is given by

$$(Lf)(p) = \int_0^\infty e^{-pt}dt = \frac{1}{p}.$$

(ii) Let $f(t) = t^n$ for $n \in \mathbb{N}$, then the associated Laplace transforms are

$$(Lf)(p) = \int_0^\infty t^n e^{-pt}dt;$$

now introduce the change of variable $s = pt$, which implies that $dt = \frac{ds}{p}$, then the Laplace transform above becomes

$$(Lf)(p) = \int_0^\infty \left(\frac{s}{p}\right)^n e^{-s}\frac{ds}{dp} = \frac{1}{p^{n+1}}\int_0^\infty s^n e^{-s}ds \equiv \frac{\Gamma(n+1)}{p^{n+1}} = \frac{n!}{p^{n+1}}.$$

(iii) Let $f(t) = e^{iat}$ for $a > 0$, then the associated Laplace transform is given by

$$\begin{aligned}(Lf)(p) &= \int_0^\infty e^{-pt}e^{iat}dt = \int_0^\infty e^{-t(p-ia)}dt,\\ &= \left[-\frac{e^{-t(p-ia)}}{(p-ia)}\right]_{t=0}^{t\to\infty} = 0 + \frac{1}{p-ia},\\ &= \frac{p+ia}{p^2+a^2},\end{aligned}$$

provided that $p > 0$ so that $e^{-p-ia} \to 0$ as $t \to \infty$. If we take the real parts of the transform above on both sides we see that the Laplace transform of $\cos(at)$ is $\frac{p}{p^2+a^2}$. In Table 12.1 we present a list of the Laplace transform for nine different common functions as a point of reference to help identify the transform that we have after solving algebraic equations.

Properties of Laplace transforms

We now consider some important properties of Laplace transforms that make them appealing to use to simplify different problems.

(i) $L(\lambda f)(p) = \lambda(Lf)(p)$.

(ii) $L(f+g)(p) = \int_0^\infty (f(t)+g(t))e^{-pt}dt = \int_0^\infty f(t)e^{-pt}dt + \int_0^\infty g(t)e^{-pt}dt = (Lf)(p) + (Lg)(p)$.

(iii) $(Lf(at))(p) = \int_0^\infty f(at)e^{-pt}dt = \int_0^\infty f(s)e^{-\frac{ps}{a}}\frac{ds}{a} = \frac{1}{a}(Lf)\left(\frac{p}{a}\right)$, where $s = at$, and we require that $a > 0$ and that $\frac{p}{a} > \alpha$.

Table 12.1 The Laplace Transform for Some Commonly Used Functions.

Function	Transform
1	$\frac{1}{p}$
t^n	$\frac{n!}{p^{n+1}}$ $(n=0,1,2,\ldots)$
e^{at}	$\frac{1}{p-a}$
$t^n e^{at}$	$\frac{n!}{(p-a)^{n+1}}$ $(n=0,1,2,\ldots)$
$\sin(at)$	$\frac{a}{p^2+a^2}$
$\cos(at)$	$\frac{p}{p^2+a^2}$
$\sinh(at)$	$\frac{a}{p^2-a^2}$
$\cosh(at)$	$\frac{p}{p^2-a^2}$
t^α	$\frac{\Gamma(\alpha+1)}{p^{\alpha+1}}$ $(\alpha>-1)$

(iv) $L\left(e^{at}f(t)\right)(p)=\int_0^\infty f(t)\,e^{-t(p-a)}dt=(Lf)(p-a)$, provided $p-a>\alpha$.

(v) $\left(Lf'\right)(p)=\int_0^\infty f'(t)\,e^{-pt}dt=\left[f(t)\,e^{-pt}\right]_{t=0}^{t\to\infty}+p\int_0^\infty f(t)\,e^{-pt}dt=-f(0)+p\,(Lf)(p)$, for $p>\alpha$.

Example 12.30. *Suppose that y satisfies the ordinary differential equation*

$$\frac{d^2y}{dx^2}+3\frac{dy}{dx}+2y=e^{-x}\quad(x\geq 0), \tag{12.104}$$

with the initial conditions $y(0)=1$ and $y'(0)=0$. Find y through the use of Laplace transforms.

We shall assume that y grows like $e^{\alpha x}$ as $x\to\infty$ for some α; we take the Laplace transform

$$\begin{aligned}(Ly)(p)&=-y(0)+p\,(Ly)(p),\\&=-1+p\,(Ly)(p),\\\left(Ly''\right)(p)&=-y'(0)+p\left(Ly'\right)(p),\\&=-p+p^2\,(Ly)(p),\\L\left(e^{-x}\right)(p)&=L\left(e^{(-1)(x)}\cdot 1\right)(p)=\frac{1}{p+1}.\end{aligned}$$

Substituting the information above into (12.104) results in

$$\begin{aligned}L\left(y''+3'+2y\right)(p)&=L\left(e^{-x}\right)(p),\\-p+p^2\,(Ly)(p)-3+3p\,(Ly)(p)+2\,(Ly)(p)&=\frac{1}{p+1},\end{aligned}$$

and collecting similar terms leads to

$$\left(p^2+3p+2\right)(Ly)(p)=p+3+\frac{1}{p+1}=\frac{p^2+4p+4}{p+1}=\frac{(p+2)^2}{p+1}.$$

Isolating $(Ly)(p)$ results in

$$\begin{aligned}(Ly)(p) &= \frac{(p+2)^2}{(p+1)^2(p+2)} = \frac{p+2}{(p+1)^2},\\ &= \frac{1}{p+1} + \frac{1}{(p+1)^2},\\ &= L\left(e^{-x}\right)(p) + L\left(xe^{-x}\right)(p),\\ \Rightarrow y(x) &= e^{-x} + xe^{-x},\end{aligned}$$

which is a solution to (12.104).

The example above shows us that some information about differential equations, especially information about the *particular integral*, can easily be obtained using Laplace transforms. The point to note about the differential equation in (12.104) is that the form of the transformed differential equation is

$$(poly)(Ly)(p) + poly = h(p),$$

where $poly$ stands for polynomial, and hence

$$(Ly)(p) = \frac{f(p)}{g(p)} + \frac{h(p)}{g(p)}, \tag{12.105}$$

where f and g are polynomials and h is the Laplace transform of a given function.

We can express $\frac{f}{g}$ as a partial fraction and hence find its inverse transformation. The second ratio, $\frac{h(p)}{g(p)}$, is slightly trickier for we do not know the form of the transform $h(p)$; however, if we split up $\frac{1}{g(p)}$ into partial fractions, we shall have a sum of terms of the form $\frac{1}{(p-a)^n}h(p)$, which is the product of two Laplace transforms.

Definition 12.31. Suppose that the function f, defined as $f : [0, \infty) \to \mathbb{C}$ is piecewise continuous and f has order $e^{\alpha t}$ as $t \to \infty$, and define the convolution of f and g, denoted as before by $f * g$, as

$$(f * g)(x) = \int_0^x f(x-y)\, g(y)\, dy.$$

Through the change of variable, $z = x - y$, it can easily be shown that $(f * g)(x) = (g * f)(x)$.

We therefore define the Laplace transform of the convolution of two functions $L(f * g)(p)$ as

$$L(f * g)(p) = (Lg)(p)(Lf)(p). \tag{12.106}$$

This all leads to Lerch's theorem:

Theorem 12.32. ***Lerch's Theorem:*** *Suppose that functions f and g, defined by $f, g : [0, \infty) \to \mathbb{C}$ are piecewise continuous and have order $e^{\alpha t}$ and $e^{\beta t}$, respectively, as $t \to \infty$, and that their Laplace transforms are equal for all $p > \gamma$, then f and g are equal at all points at which both functions are continuous.*

Example 12.33. *Let y satisfy the initial value ordinary differential equation*

$$\frac{d^2y}{dt^2} + a\frac{dy}{dt} + by = f(t) \quad (t \geq 0), \tag{12.107}$$

where $y(0) = 1$ and $y'(0) = 0$, where a and b are constants. By using the theory of Laplace transforms, find y.

Solution.

$$\begin{aligned}\left(Ly'\right)(p) &= -y(0) + p\,(Ly)(p)\,,\\ &= -1 + p\,(Ly)(p)\,,\\ \left(Ly''\right)(p) &= -y'(0) + p\left(Ly'\right)(p)\,,\\ &= -p + p^2\,(Ly)(p)\,.\end{aligned}$$

Therefore, we have

$$\left(p^2 + ap + b\right)(Ly)(p) - a - p = (Lf)(p)\,,$$

$$(Ly)(p) = \frac{p+a}{p^2+ap+b} + \frac{(Lf)(p)}{p^2+ap+b}.$$

Suppose that the quadratic equation, $\lambda^2 + a\lambda + b = 0$, has two distinct solutions, α and β, so that $p^2 + ap + b = (p-\alpha)(p-\beta)$, and that $\frac{1}{p^2+ap+b} = \frac{1}{\alpha-\beta}\left(\frac{1}{p-\alpha} - \frac{1}{p-\beta}\right)$. This then results in

$$(Ly)(p) = \frac{1}{\alpha-\beta}\left(\frac{\alpha+a}{p-\alpha} - \frac{\beta+a}{p-\beta}\right) + \frac{1}{\alpha-\beta}\left(\frac{1}{p-\alpha} - \frac{1}{p-\beta}\right)(Lf)(p)\,. \tag{12.108}$$

We already know that $\frac{1}{p-\alpha}$ is the Laplace transform of $e^{\alpha t}$ and as such $\frac{1}{\alpha-\beta}\left(\frac{1}{p-\alpha} - \frac{1}{p-\beta}\right)$ is the Laplace transform of $\frac{1}{\alpha-\beta}\left(e^{\alpha t} - e^{-\beta t}\right)$, so that the last term for $(Ly)(p)$ from above is the transform of the convolution of $\frac{1}{\alpha-\beta}\left(e^{\alpha t} - e^{-\beta t}\right)$ and $f(t)$. This is to say,

$$\int_0^t \frac{e^{\alpha(t-s)} - e^{-\beta(t-s)}}{\alpha-\beta} f(s)\,ds.$$

This implies that our solution for y is given by

$$y(t) = \frac{\alpha-a}{\alpha-\beta}e^{\alpha t} - \frac{\beta+a}{\alpha-\beta}e^{\beta t} + \int_0^t \phi(t-s)\,f(s)\,ds, \tag{12.109}$$

where

$$\phi(x) = \frac{e^{\alpha x} - e^{\beta x}}{\alpha-\beta}.$$

As an aside, we should note that the function ϕ satisfies the homogeneous differential equation and the initial conditions $\phi(0) = 0$ and $\phi'(0) = 1$.

If we had the case where $p^2 + ap + b = 0$ has a double root so that $p^2 + ap + b = (p-\alpha)^2$, then the Laplace transform would become

$$\begin{aligned}(Ly)(p) &= \frac{p+a}{(p-\alpha)^2} + \frac{1}{(p-\alpha)^2}(Lf)(p)\,,\\ &= \frac{1}{(p-\alpha)} + \frac{a+\alpha}{(p-\alpha)^2} + \frac{1}{(p-\alpha)^2}(Lf)(p)\,,\end{aligned}$$

so looking up the inverse transforms, we see that

$$\begin{aligned} y(t) &= e^{\alpha t} + (a+\alpha) t e^{\alpha t} + \left(t e^{\alpha t}\right) * f, \\ &= e^{\alpha t} + (a+\alpha) t e^{\alpha t} + \int_0^x \phi(t-s) f(s)\, ds, \end{aligned}$$

where $\phi(x) = x e^{\alpha x}$, that is to say, ϕ satisfies the homogeneous differential equation with the initial conditions $\phi(0) = 0$ and $\phi'(0) = 1$.

Example 12.34. *The current y in an electrical circuit obeys the equation*

$$\frac{d^2 y}{dt^2} + a\frac{dy}{dt} + by(t) = \sin \omega t, \tag{12.110}$$

where $a \geq 0$, $b, \omega > 0$ and are constants, and we have the initial conditions $y(0) = 0$, and $y'(0) = 1$. Show that if $a > 0$ then the solutions y remains bounded $\forall t > 0$, while if $a = 0$ then the solution is bounded if $\omega \neq \sqrt{b}$.

Solution. We have already derived the Laplace transform of the left-hand side of (12.110), and we know that $L(\sin \omega t)(p) = \frac{\omega}{p^2+\omega^2}$, which then implies that we are solving

$$(Ly)(p) = \frac{1}{p^2 + ap + b} + \frac{1}{p^2 + ab + b}\frac{\omega}{p^2 + \omega^2}. \tag{12.111}$$

If we have $\lambda^2 + a\lambda + b = 0 \Leftrightarrow \lambda = -\frac{1}{2}a \pm \frac{\sqrt{a^2-4b}}{2}$ and if $b > 0$ then either λ has two real roots which are positive $\left(a^2 - 4b \geq 0\right)$ for then $\left|a^2 - 4b\right| < \left|a^2\right|$ or there are two complex roots with real parts $-\frac{1}{2}a$.

In either case, if $a > 0$, then the roots of $\lambda^2 + a\lambda + b$ have negative real parts, which we shall call α and β, that they may not be distinct, so that $p^2 + ap + b = (p-\alpha)(p-\beta)$. This then makes

$$\begin{aligned} (Ly)(p) &= \frac{1}{(p-\alpha)(p-\beta)} + \frac{1}{(p-\alpha)(p-\beta)}\frac{\omega}{p^2+\omega^2}, \\ &= \left(\frac{A}{(p-\alpha)} + \frac{B}{(p-\beta)}\right) + \left(\frac{A}{(p-\alpha)} + \frac{B}{(p-\beta)}\right)\frac{\omega}{p^2+\omega^2}, \end{aligned} \tag{12.112}$$

for constants A and B. This then implies that the solution to (12.110) is

$$y(t) = Ae^{\alpha t} + Be^{\beta t} + A\left(e^{\alpha t} * \sin \omega t\right) + B\left(e^{\beta t} * \sin \omega t\right). \tag{12.113}$$

Since the real component of α is less than zero, we have that $Ae^{\alpha t}$ is bounded as $\to \infty$ and so is $Be^{\beta t}$. Now we turn our attention to the particular integral equivalent component of (12.113), where we have

$$\begin{aligned} \left|\left(e^{\alpha t} * \sin \omega t\right)\right| &= \int_0^t \left|e^{\alpha(t-s)} \sin \omega s\right| ds, \\ &\leq \int_0^t \left|e^{\alpha(t-s)}\right| ds, \quad \text{since } |\sin \omega t| \leq 1, \\ &\leq \int_0^t e^{Re(\alpha)(t-s)} ds, \end{aligned}$$

$$= \frac{e^{Re(\alpha)t} - 1}{Re\,(\alpha)},$$

which is a bounded function. So if $a > 0$, then y is bounded.

If we consider the case where $a = 0$, then

$$(Ly)\,(p) = \frac{1}{p^2 + b} + \frac{1}{p^2 + b}\frac{\omega}{p^2 + \omega^2}. \tag{12.114}$$

If $b \neq \omega^2$, then we can split the second term above into partial factions as

$$\frac{A}{p^2 + b} + \frac{B}{p^2 + \omega^2}.$$

Thus we have a solution of the form

$$y = \frac{1}{\sqrt{b}}\sin\left(\sqrt{b}t\right) + \frac{A}{\sqrt{b}}\sin\left(\sqrt{b}t\right) + \frac{B}{\omega}\sin\left(\omega t\right),$$

which is again a bounded function as $t \to \infty$.

However, if $b = \omega^2$ then we have $\frac{\omega}{p^2+\omega^2} = \frac{1}{p^2+b} = \frac{\omega}{\left(p^2+\omega^2\right)^2}$ which is $\left(\frac{\omega}{p^2+\omega^2}\frac{\omega}{p^2+\omega^2}\right)\frac{1}{\omega}$, and is the Laplace transform of the convolution

$$\begin{aligned}\frac{1}{\omega}\left(\sin\left(\omega t\right) * \sin\left(\omega t\right)\right) &= \frac{1}{\omega}\int_0^t \sin\left(\omega\left(t - s\right)\right)\sin\left(\omega s\right) ds,\\ &= \frac{1}{2\omega}\int_0^t \left(\cos\left(\omega\left(t - 2s\right)\right) - \cos\left(\omega\right)\right) ds,\\ &= \frac{1}{2\omega}\left(\frac{2\sin\left(\omega t\right)}{2\omega} - t\cos\left(\omega t\right)\right),\end{aligned}$$

so the solution for this situation is

$$y = \frac{1}{\sqrt{b}}\sin\left(\sqrt{b}t\right) + \frac{\sin\left(\omega t\right)}{2\omega^2} - \frac{1}{2\omega}t\cos\left(\omega t\right),$$

which is not a bounded function.

The purpose of this section on Fourier methods and Laplace transforms is to introduce the theory that is the basis of **spectral modeling** of partial differential equations on the sphere. Therefore, we now move on to provide an introduction to spectral modeling, which is used extensively in numerical weather prediction.

12.5 Spectral Modeling

In Chapters 8 and 9 we introduced the finite difference methods which only provide information of our solution at the individual grid points, and as such no information about the solution between the points is provided. An alternative approach is to expand the dependent variables in terms of a finite series of smooth orthogonal functions. As a result of assuming this form for the dependent variables, the problem

is then reduced to solving a set of ordinary differential equations that determine the behavior in time of the expansion coefficients.

We start with a linear one-dimensional evolutionary differential equation given by

$$\frac{\partial \psi}{\partial t} = \mathrm{L}(\psi), \tag{12.115}$$

where L is a linear differential operator. The spectral approach is to assume that we can expand the function, ψ, in terms of a set of orthogonal functions, denoted by $e_k(x)$, where $k = k_1, \ldots, k_n$, such that

$$\psi(t, x) = \sum_k \psi_k(t)\, e_k(x), \tag{12.116}$$

where the ψ_k terms are the expansion coefficients that we need to determine. For the expression in (12.116) there are two possible techniques that could be used. The first technique is based upon minimizing a *residual* in a least squares formulation. Here we shall present the general expression where the functions e_k are linearly independent functions. The first step is to substitute (12.116) into (12.115) and define the residual as

$$\mathcal{R} = \sum_{k_1}^{k_2} \frac{\psi_k}{dt} e_k - \sum_k \psi_k \mathrm{L}(e_k). \tag{12.117}$$

The next step is to choose the time derivative $\frac{d\psi_k}{dt}$ by minimizing the residual. This minimization can be achieved through a least square approach:

$$\mathcal{I} = \int \mathcal{R}^2 dx, \tag{12.118}$$

with respect to the time derivative. Therefore, we have

$$\begin{aligned}\int \mathcal{R}^2 dx &= \int \left(\sum_k \frac{d\psi_k}{dt} e_k - \sum_k \psi_k \mathrm{L}(e_k)\right)\left(\sum_l \frac{d\psi_l}{dt} e_l - \sum_l \psi_l \mathrm{L}(e_l)\right) dx, \\ &= \int \left(\frac{d\psi_k}{dt}\right)\left(\frac{d\psi_l}{dt}\right) e_k e_l - \frac{d\psi_k}{dt}\psi_l \mathrm{L}(e_l) - \frac{d\psi_l}{dt}\psi_k \mathrm{L}(e_k) + \psi_k \psi_l \mathrm{L}(e_k)\mathrm{L}(e_l)\, dx. \end{aligned} \tag{12.119}$$

We now differentiate (12.119) with respect to $\frac{d\psi_k}{dt}$ and set the derivative equal to zero, which results in

$$\sum_l \frac{\partial \psi_l}{dt} \int e_k e_l dx = \sum_l \int e_k H(e_l)\, dx, \quad k = k_1, \ldots, k_2. \tag{12.120}$$

The second approach is referred to as the **Galerkin method**, where we seek an approximation so that the **residual is orthogonal** to the space from which $\psi(x, t)$ comes from. We start by setting

$$\int \mathcal{R}\psi_i dx = 0, \quad i = 1, 2, \ldots, N+1, \tag{12.121}$$

where ψ_i can be any set of linearly independent test functions. If the expansion functions are used as test functions, then we obtain (12.120). Since the expansion functions are known, (12.116) can be used

to provide the expansion coefficients ψ_k, given the grid point values ψ_l. The integral in (12.120) can be calculated exactly for all possible values of k and l. This implies that (12.120) reduces to a set of coupled ordinary differential equations that can be solved for the $\frac{d\psi_k}{dt}$ given the values of ψ_k. This leads to the complete solution as

$$\frac{\partial \psi}{\partial t} = \sum_k \frac{d\psi_k}{dt} e_k. \tag{12.122}$$

Returning to the spectral approach where e_k are orthogonal, we have that the property of the expansion functions are orthonormal so that they satisfy the conditions

$$\int_0^L \overline{e_l} e_k dx = \begin{cases} 1 & l = k \\ 0 & l \neq k \end{cases}, \tag{12.123}$$

where $\overline{e_l}$ is the complex conjugate of e_l. Applying this condition above, we obtain

$$\frac{d\psi_k}{dt} = \sum_l \psi_l \int_0^L \overline{e_k} \mathrm{L}(e_l)\, dx, \quad \forall k. \tag{12.124}$$

That is to say that we now have a set of ordinary differential equations for the rate of change with time of the expansion coefficient.

We now consider how the choice of expansion functions can greatly simplify the problem of interest.

(a) If the expansion functions are eigenfunctions of L, then we have that $\mathrm{L}(e_l) = \lambda_l e_l$, where the λ_l are the eigenvalues. This makes (12.124), $\frac{d\psi_l}{dt} = \lambda_l \psi_l$.

(b) If the original ordinary differential equation is of the form

$$\mathrm{M}\left(\frac{\partial \psi}{\partial t}\right) = \mathrm{L}(\psi),$$

where M is a linear operator, then the problem is simplified by using the eigenfunctions of M with eigenvalues λ_m as expansion functions so that we obtain

$$\lambda_m \frac{d\psi}{dt} = \sum_l \psi_l \int_0^L \overline{e}_m \mathrm{L}(e_l)\, dx. \tag{12.125}$$

One-dimensional linear advection equation example

We can write the advection equation in terms of the longitude λ, and angular velocity $\gamma = \frac{2\pi u_0}{L}$, where u_0 is the advection wind speed, such that the partial differential equation as

$$\frac{\partial \varphi}{\partial t} + \gamma \frac{\partial \varphi}{\partial \lambda} = 0, \tag{12.126}$$

that has the periodicity boundary condition, $\varphi(\lambda, t) = \varphi(\lambda + 2\pi K, t)$, for some integer K, and initial condition, $\varphi(\lambda, t) = f(\lambda)$.

Given the partial differential equation in (12.126), and if we are going to apply a series expansion approximation for the solution to (12.126), then we have to choose a suitable set of expansion functions.

The most common choice is to apply a finite Fourier series approximation in the form

$$\varphi(\lambda, t) = \sum_{m=-M}^{M} \varphi_m(t) e^{im\lambda}, \tag{12.127}$$

because the expansion functions are the eigenfunctions of the spatial differential operator. The number M is the maximum wavenumber and the φ_m are the complex expansion coefficients. We also have the property that $\varphi_{-m}(t) = \overline{\varphi_m(t)}$, and as such we only have to solve for the φ_m for $0 \leq m \leq M$.

If we substitute (12.127) into (12.126), then equating coefficients of the expansion factors yields

$$\frac{d\varphi_m}{dt} + im\gamma\varphi_m = 0, \quad 0 \leq m \leq M, \tag{12.128}$$

which then results in $2M + 1$ equations for the φ_m. It is possible to integrate (12.128) exactly, which results in

$$\varphi_m = \varphi_m(0) e^{im\gamma t}, \quad 0 \leq m \leq M. \tag{12.129}$$

We now have to consider the initial conditions for φ. If we approximated this function with a truncated Fourier series with respect to just the spatial coordinate by

$$f(\lambda) = \sum_{m=-M}^{M} a_m e^{im\lambda},$$

then we would obtain the complete solution in the form

$$\varphi(\lambda, t) = \sum_{m=-M}^{M} a_m e^{im(\lambda - \gamma t)}. \tag{12.130}$$

An important property of the solution in (12.130) is that it does not suffer from the dispersion problem that plagues some finite difference methods due to the spatial derivative being calculated analytically and not approximated which is the case for the finite difference schemes. However, there are certain types of geophysical problems that involve nonlinear differential equations to solve and as such we now look at the nonlinear advection equation to ascertain what changes are needed when considering these types of problems.

If we now multiply (12.127) by each of the basic functions and apply the orthogonality property of Fourier basis, then at the initial time we obtain

$$\varphi(0) = A_m \int_0^{2\pi} \varphi(\lambda, 0) e^{-im\lambda} d\lambda, \tag{12.131}$$

where A_m are the normalization factors. We should note that we can apply (12.127) to obtain the space distribution of the solution at any future time. This process is an application of the inverse Fourier transform.

It is often the case that the initial conditions are given at a set of $N + 1$ grid point that have a spacing of Δx. This means that the truncated Fourier series could be interpret as representing an interpolating

function that exactly fits the values of φ at the $N+1$ grid points. Therefore, it is possible to compute (12.131) as a discrete sum as

$$\varphi(0) = A'_m \sum_{i=1}^{K} \varphi(\lambda_i) e^{-im\lambda_i}, \tag{12.132}$$

which we have already seen is the discrete Fourier transform. The corresponding inverse Fourier transform for this setup is defined as

$$\varphi(\lambda_i, 0) = \sum_{m=-M}^{M} \varphi_m(0) e^{im\lambda_i}. \tag{12.133}$$

Both of the expressions in (12.132) and (12.133) are calculated through the fast Fourier transforms. It can be shown that if we start from the set of $\varphi_m(0)$, and then move to the set of $\varphi(\lambda_i, 0)$ for $i = 1, 2, \ldots, K$ and return to $\varphi_m(0)$, then we recover the original values as long as we have that $K \geq 2M+1$ and that the associated grid points are equally spaced. The distribution of these $2M+1$ points is referred to as the **linear grid**.

However, it can be shown that when we have the product of two functions, it is possible to avoid aliasing errors as long as we have minimum of $K \geq 3M+1$ grid points. The distribution of these points is referred to as the **quadratic grid**.

If we consider now the time component, given the derived initial conditions for the spectral coefficients, then we have to integrate the ordinary differential equation for the expansion coefficients to some future time. This goal is usually achieved through some form of time-stepping finite difference procedure.

Nonlinear advection equation

The nonlinear advection equation is written in the form

$$\frac{\partial \varphi}{\partial t} = -\varphi \frac{\partial \varphi}{\partial \lambda}. \tag{12.134}$$

We shall again assume a truncated Fourier series as our estimate of a solution to (12.134) as defined in (12.127), which implies that the right-hand side of (12.134) becomes

$$F = \sum_{m=-2M}^{2M} F_m e^{im\lambda} \text{ where } F_m = i \sum_{\hat{m}=m-M}^{M} \left(m - \hat{m}\right) \varphi_{\hat{m}} \varphi, \quad \text{for } m \geq 0. \tag{12.135}$$

Substituting (12.127) into the left-hand side of (12.134) results in

$$\frac{\partial \varphi}{\partial t} = \sum_{m=-M}^{M} \frac{d\varphi_m}{dt} e^{im\lambda}. \tag{12.136}$$

A feature that we have to note here is that the two series presented above as approximations to (12.134) are summations truncated at two different wavenumbers, and as such there will always be a residual associated with this approach. Given that there is a residual, denote by $\mathcal{R}$ again, we will select the $\frac{d\varphi_m}{dt}$ terms such that they satisfy the condition

$$\int_0^{2\pi} \mathcal{R} e^{im\lambda} d\lambda = 0, \quad \forall m,$$

where this condition is the results of applying a Galerkin method to the residuals. It can be shown that the required time derivatives are

$$\frac{d\varphi_m}{dt} = F_m, \quad \text{for } -M \leq m \leq M.$$

This means that the Fourier components, F_m, with wavenumbers that are larger than M are neglected. The implication of this is that there is no aliasing of small-scales components outside of the original truncation, which implies that there is no nonlinear instability.

Before we progress, we shall quickly explain what is meant by **aliasing error**.

Definition 12.35. Aliasing is an effect that causes different signals to become indistinguishable when sampled. That is to say that the two signals are aliases of each other. The aliasing error is defined as the distortion of an original continuous signal, or wave, if reconstructed from samples that are from different signal/waves than the original continuous signal/wave.

As an example of where aliasing could occur, we have plotted two sine waves in Fig. 12.19 that have different frequencies that pass through the same sample points. We can see that if we reconstructed the wave from the larger frequency, then we would not be reconstructing the oscillation associated

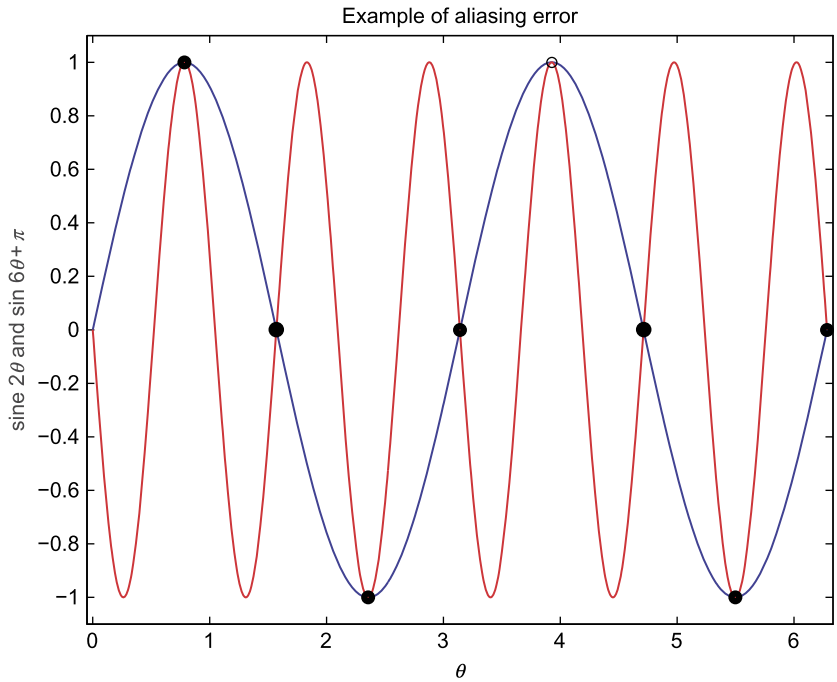


FIGURE 12.19

Plot of an example of an aliasing error where both sine waves match the observation points.

with the faster wave correctly and hence in a numerical modeling situation we would be incorrectly reconstructing the smaller-scale features.

Returning to the nonlinear advection example, we now address a technique used to calculate the nonlinear term $\varphi \frac{\partial \varphi}{\partial x}$ with spectral methods, referred to as the **transform method**, which involves the use of transforms, especially **discrete Fourier transform**, and then the **fast Fourier transforms**, which we introduce next.

Discrete Fourier transform

The discrete Fourier transform is the equivalent of the continuous Fourier transform for signals that are known at only N points, and are separated by Δx in space. If we let $f(x)$ be the continuous function which is the source of the data, let N samples be denoted as $f(x_0), f(x_1), \ldots, f(k), \ldots f(x_{N-1})$, and recall that the Fourier transform of the original function $f(x)$ would be

$$F(t) = \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i x t} dx, \tag{12.137}$$

then each of the grid points contain a value of the function $f(k)$ representing an area. Since the integrand exists only at the grid point, we have

$$\begin{aligned} F(t) &= \int_0^{(N-1)\Delta x} f(x)\, e^{-2\pi i x t} dx, \\ &= f(x_0)\, e^{-i0} + f(x_1)\, e^{-2\pi i \Delta x} + f(x_2)\, e^{-2\pi i 2\Delta x} + \cdots + f(k)\, e^{-2\pi i k \Delta x} + \cdots + f\left(x_{n-1}\right) e^{-2\pi i (N-1)\Delta x}, \\ &= F(t) = \sum_{k=0}^{N-1} f(x_k)\, e^{-2\pi i k \Delta x}. \end{aligned} \tag{12.138}$$

The inverse discrete Fourier transform can easily be shown to be

$$f(x) = \frac{1}{N} \sum_{n=0}^{N-1} F(x_n)\, e^{2\pi i n \Delta x}. \tag{12.139}$$

Fast Fourier transforms

As the name of this subsection suggests, there have been developments over the last 60 years to find ways to be able to perform the multiplications required for the discrete Fourier transform and its inverse so that the number of operations is drastically reduced. This enables the movement between the grid point space and the frequency space to be quick and efficient, but it also allows for a more practical implementation of the spectral model in the modern-day supercomputer infrastructure.

The fast Fourier transform reduces the number of computation needed for N points from $2N^2$ to $2N \log_2 N$, where $\log_2$ is the base 2 logarithm. Fast Fourier transforms were first officially discussed in a paper by Cooley and Tukey [71]. However, there is evidence that Gauss derived a similar formulation in 1805.

The definition of a fast Fourier transform is that it computes the discrete Fourier transform and produces exactly the same result as evaluating the discrete Fourier transform definition directly, but as the name suggests, these algorithm are faster than the direct evaluation. The algorithm for the fast Fourier transform from [71] takes the discrete Fourier transform and first rearranges the input data into

bit-reversed order, then it builds the output transform. The basic idea of the Cooley and Tukey algorithm is to break up the transform of length N into two transforms of length $\frac{N}{2}$ using the identity

$$\begin{aligned}\sum_{n=0}^{N-1} a_n e^{\frac{-2\pi i n k}{N}} &= \sum_{n=0}^{\frac{N}{2}-1} a_{2n} e^{\frac{-2\pi i (2n) k}{N}} + \sum_{n=0}^{\frac{N}{2}-1} a_{2n+1} e^{\frac{-2\pi i (2n+1) k}{N}}, \\ &= \sum_{n=0}^{\frac{N}{2}-1} a_n^{even} e^{\frac{-2\pi i (2n) k}{N}} + e^{-\frac{2\pi i k}{N}} \sum_{n=0}^{\frac{N}{2}-1} a_n^{odd} e^{\frac{-2\pi i (2n) k}{N}}. \end{aligned} \tag{12.140}$$

The reason for introducing the discrete and the fast Fourier transforms is because of the application in spectral modeling. We start this subsection considering the nonlinear advection equation, and so we now explain why we require these transforms.

The advantage of the fast Fourier transform is that it is easy to move from the spectral representation (spectral space) to a grid-point representation, often referred to as physical space. The essence of the transform method is to calculate derivatives in spectral space, but to transform to physical space using the fast Fourier transform whenever a product is required. Once all of the products have been computed at grid points, the spectral coefficients of this product fields are calculated.

Therefore, given the φ_m we wish to compute the spectral coefficients of the nonlinear term $-\varphi \frac{\partial \varphi}{\partial \lambda}$, which is F_m in (12.135). For us to achieve this goal we have to follow the following three-step process:

1. Calculate the functions φ and $\frac{\partial \varphi}{\partial \lambda}$ at the grid points λ_i by using the spectral coefficients

$$\varphi(\lambda_i) = \sum_m \varphi_m e^{im\lambda_i}, \quad \frac{\partial \varphi}{\partial \lambda_i} = \sum_m im\varphi_m e^{im\lambda_i}.$$

2. Calculate the advection term at each grid point in physical space

$$F(\lambda_i) = -\varphi(\lambda_i) \frac{\partial \varphi}{\partial \lambda_i}.$$

3. Finally, this product is returned to spectral space through calculating the Fourier coefficients

$$F_m = \frac{1}{2\pi} \sum_i F(\lambda_i) e^{-im\lambda_i}.$$

We should note here that the procedure described above has to be employed to calculate the spectral coefficient of the nonlinear advection term at each time step. Another feature to note is that as the product of two functions is computed in grid-point space, not spectral space, we will have aliasing unless the number of grid points corresponds to the quadratic grid.

We have shown the basis for spectral method here, but we now move on to present the theory that will enable us to extend the spectral methods to spherical coordinates.

12.5.1 Sturm-Liouville Theory

Another important theory that we need to introduce in order to understand the implementation of the spectral method is the Sturm-Liouville theory. The **Sturm-Liouville equation** is a real second-order

linear differential equation of the form

$$\frac{d}{dx}\left(p(x)\frac{dy}{dx}\right) = q(x) = -\lambda w(x) y, \tag{12.141}$$

where the functions $p(x)$, $q(x)$ and $w(x)$ are specified. In the simplest case all the coefficients are continuous on the finite closed interval $[a, b]$, the function, $p(x)$, has a continuous first derivative, and we have boundary conditions on the solution $y(x)$ at $x = a$ and $x = b$. The value of λ is not specified in the equation and as such finding the values for λ, so that there exists non-trivial solutions of (12.141) is part of the set of problems referred to as **Sturm-Liouville** problems.

A Sturm-Liouville problem is said to be regular if $p(x)$, $w(x) > 0$, $p(x)$, $p'(x)$, $q(x)$ and $w(x)$ are continuous functions over the finite interval $[a, b]$, and has separated boundary conditions of the form

$$\alpha_1 y(a) + \alpha_2 y'(a) = 0 \quad \left(\alpha_1^2 + \alpha_2^2 > 0\right), \tag{12.142a}$$

$$\beta_1 y(b) + \beta_2 y'(b) = 0 \quad \left(\beta_1^2 + \beta_2^2 > 0\right). \tag{12.142b}$$

We now introduce the definition of a **regular singular point**.

Definition 12.36. Consider the second-order ordinary differential equation of the form

$$y'' + P(x) y' + Q(x) y = 0. \tag{12.143}$$

If $P(x)$ and $Q(x)$ remain finite at $x = x_0$, then x_0 is called an **ordinary point**. If either $P(x)$ or $Q(x)$ diverges as $x \to x_0$, then x_0 is called a **singular point**. If either $P(x)$ or $Q(x)$ diverges as $x \to x_0$, but $(x - x_0) P(x)$ and $(x - x_0)^2 Q(x)$ remains finite as $x \to x_0$, then the point $x = x_0$ is called a **regular singular point**.

Given the assumption that the Sturm-Liouville problem is regular, then there are the following properties:

- The eigenvalues $\lambda_1, \lambda_2, \ldots$ of the regular Sturm-Liouville problem defined by (12.141), (12.142a), and (12.142b) are real and are able to be ordered such that

$$\lambda_1 < \lambda_2 < \cdots < \lambda_n < \cdots \to \infty.$$

- For each eigenvalue, λ_n there is a corresponding unique eigenfunction $y_n(x)$ that has exactly $n - 1$ zeros in the interval (a, b). The eigenfunction $y_n(x)$ is referred to as the nth fundamental solution satisfying the regular Sturm-Liouville problem given by (12.141), (12.142a), and (12.142b).
- The normalized eigenfunctions form an orthonormal basis

$$\int_a^b y_n(x) y_m(x) w(x) dx = \delta_{mn} = \begin{cases} 1 & m = n \\ 0 & m \neq 0 \end{cases}. \tag{12.144}$$

12.5.2 Legendre Differential Equation

If we consider the basic spherical coordinate system, then we have a point P, that is specified by the distance r from the origin, the angle θ between the position vector and the z axis, and the angle ϕ from

the x axis to the projection of the position vector onto the xy plane. We have presented an illustration of this coordinate system in Fig. 12.20.

The Laplacian equation for a function $F(r, \theta, \phi)$ is given by

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial F}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial F}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2 F}{\partial\lambda^2} = 0. \tag{12.145}$$

If we only consider solutions that are dependent on r and θ, then (12.145) simplifies to

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial F}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial F}{\partial\theta}\right) = 0. \tag{12.146}$$

We now assume that it is possible to apply the separation of variables technique to solve (12.146). This implies that we are seeking a solution of the form $Y(r, \theta) = R(r)\,\Theta(\theta)$. Assuming a solution of this form results in the following two uncoupled ordinary differential equations to solve:

$$r^2\frac{d^2R}{dr^2} + 2r\frac{dR}{dr} - \lambda R = 0, \tag{12.147a}$$

$$\frac{d}{d\theta}\left(\sin\theta\frac{d\Theta}{d\theta}\right) + \lambda\sin\theta\Theta = 0, \tag{12.147b}$$

where λ is referred to as the **separation constant**. The ordinary differential equation for r is equidimensional and thus has solutions that are easily found that are powers of r. The ordinary differential equation for θ is **Legendre's equation**, which can be solved through the following means.

We introduce the change of variable $\tau = \cos\theta$, and substitute this information into (12.147b), but first we recognize that $\frac{d\tau}{d\theta} = -\sin\theta$ and apply the chain rule $\frac{d\Theta}{d\tau} = \frac{d\Theta}{d\tau}\frac{d\tau}{d\theta}$, which results in

$$\begin{aligned}
\frac{d}{d\tau}\left(\frac{d\tau}{d\theta}\right)\left(\sin\theta\frac{d\tau}{d\theta}\frac{d\Theta}{d\tau}\right) + \lambda\sin\theta\Theta &= 0,\\
\frac{d}{d\tau}(-\sin\theta)\left((\sin\theta)(-\sin\theta)\frac{d\Theta}{d\tau}\right) + \lambda\sin\theta\Theta &= 0,\\
\frac{d}{d\tau}\left(\sin^2\theta\frac{d\Theta}{d\tau}\right) + \lambda\Theta &= 0,
\end{aligned}$$

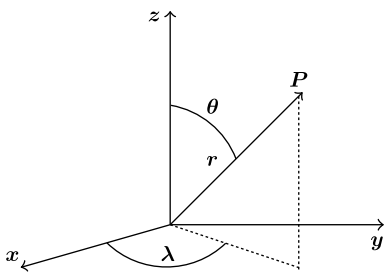


FIGURE 12.20

Schematic of the spherical coordinate system.

$$\frac{d}{d\tau}\left(\left(1-\cos^2\theta\right)\theta\frac{d\Theta}{d\tau}\right)+\lambda\Theta=0,$$
$$\frac{d}{d\tau}\left(\left(1-\tau^2\right)\theta\frac{d\Theta}{d\tau}\right)+\lambda\Theta=0, \tag{12.148}$$

where (12.148) has regular single points at the end points $\tau=\pm1$. The ordinary differential equation in (12.148) combined with the boundary condition just mentioned makes this problem in the form of a singular Sturm-Liouville equation, which we presented in the last subsection. Therefore, there is a class of λ for which there are well-behaved solutions to the eigenvalues of this problem.

We now turn our attention to method for solving (12.148); we do so by considering a power series solution for Θ about $\tau=0$ so that

$$\Theta(\tau)=\sum_{n=0}^{\infty}a_n\tau^n. \tag{12.149}$$

By substituting (12.149) into (12.148), and using the fact that the derivatives of a power series are defined as

$$\Theta'(\tau)=\sum_{n=1}^{\infty}a_n n\tau^{n-1}=\sum_{n=0}^{\infty}a_{n+1}(n+1)\tau^n, \tag{12.150a}$$
$$\Theta^{'}(\tau)=\sum_{n=2}^{\infty}a_n n(n-1)\tau^{n-2}=\sum_{n=0}^{\infty}a_{n+2}(n+1)(n+2)\tau^n, \tag{12.150b}$$

and substituting (12.150a) and (12.150b) into (12.148), we can derive a recursive relationship between a_{n+2} and a_n as follows:

$$\Theta''-\tau^2\Theta''-2\tau\Theta'+\lambda\Theta=0,$$
$$\sum_{n=2}^{\infty}a_n n(n-1)\tau^{n-2}-\sum_{n=2}^{\infty}a_n n(n-1)\tau^n-2\sum_{n=1}^{\infty}a_n\tau^n+\lambda\sum_{n=0}^{\infty}a_n\tau^n=0,$$
$$(n+1)(n+2)a_{n+2}-(n(n-1)+2n-\lambda)a_n=0,$$
$$\Rightarrow a_{n+2}=\frac{n(n+1)-\lambda}{(n+1)(n+2)}a_n. \tag{12.151}$$

For very large values of n we may approximate the recursive relationship in (12.151) by ignoring λ compared to $n(n+1)$, which simplifies the relationship to $(n+2)a_{n+2}\approx na_n$. From this it is possible to show that any such solution will be logarithmically singular at one or both end points. The only way to avoid this is for the recurrence relationship in (12.151) to terminate. We see that for this to happen we require the numerator to be equal to zero. This occurs when $\lambda=k(k+1)$ for some value of k which is a non-negative integer.

Given the termination condition, we now have a terminating solution of the form of polynomial of degree k. These types of solutions are referred to as **Legendre polynomials**.

12.5.3 Legendre Polynomials

The set of polynomials we introduce here play an important part in defining the solutions to the spherical harmonic differential equation, which then plays an important part in the application of spectral methods for modeling ordinary and partial differential equations in spherical coordinates.

We start with the Legendre functions that are the solutions to the Legendre differential equations

$$\frac{d}{dx}\left(\left(1-x^2\right)\frac{d}{dx}P_n\left(x\right)\right)+n\left(n+1\right)P_n\left(x\right)\equiv\left(1-x^2\right)\frac{d^2P_n}{dx^2}-2x*\frac{dP_n\left(x\right)}{dx}+n\left(n+1\right)=0, \tag{12.152}$$

which we have shown is a Sturm-Liouville problem. We have determined the constants $n\left(n+1\right)$ as the necessary conditions to enable the solution to (12.152) to have a regular solution in the interval $[1, 1]$. We have seen that the solution to (12.152) is a polynomial of order (n). An important property of the solution to (12.152) is that they satisfy the orthogonality relationship

$$\int_{-1}^{1}P_n\left(x\right)P_m\left(x\right)dx=\begin{cases}1 & n=m\\ 0 & n\neq m\end{cases}. \tag{12.153}$$

A second condition that the Legendre polynomials satisfy is

$$\int_{-1}^{1}P_n\left(x\right)^2dx=\frac{2}{2n+1}.$$

There are two different methods for generating the Legendre polynomials. The first method is through a generating function given by

$$G\left(x,t\right)=\sum_{n=0}^{\infty}t^nP_n\left(x\right)=\frac{1}{\sqrt{1-2xt+t^2}}. \tag{12.154}$$

The second approach for generating the polynomials is through **Rodrigue's formula**, which is given by

$$P_n\left(x\right)=\frac{1}{2^nn!}\frac{d^n}{dx^n}\left(x^2-1\right)^n. \tag{12.155}$$

Given the two expression for generating the Legendre polynomials in (12.154) and (12.155), it is possible to derive recursive relationships between the polynomials and their derivatives

$$\begin{aligned}&P'_{n+1}-P'_{n-1}=\left(2n+1\right)P_n,\\ &\left(n+1\right)P_{n+1}=\left(2n+1\right)xP_n+nP_{n-1}=0.\end{aligned}$$

To help illustrate the mathematical expressions for the Legendre polynomials, we present the first six polynomials in Table 12.2. We can see from this table that if n is even, then the associated Legendre polynomial is only a function of the even powers of x, while if n is an odd number, then the polynomials are only a function of the odd powers of x. A visualization of the first six Legendre polynomials is presented in Fig. 12.21.

An interesting feature of the plots of the Legendre polynomials in Fig. 12.21 is that each polynomial has the same number of zeros as the order of the polynomial. That is to say for P_0 then this polynomial never takes the value zero, for P_1 as one value for x where the polynomial has the value zero, while for P_2 there two values for $x\in[-1, 1]$ where $P_2\left(x\right)=0$, and as such we see that this is true for the other three polynomials presented in Fig. 12.21.

Table 12.2 Expressions for the First Six Legendre Polynomials.

Order	Polynomial
$P_0(x)$	1
$P_1(x)$	x
$P_2(x)$	$\frac{1}{2}\left(3x^2-1\right)$
$P_3(x)$	$\frac{1}{2}\left(5x^3-3x\right)$
$P_4(x)$	$\frac{1}{8}\left(35x^4-30x^2+3\right)$
$P_5(x)$	$\frac{1}{8}\left(63x^5-70x^3+15x\right)$

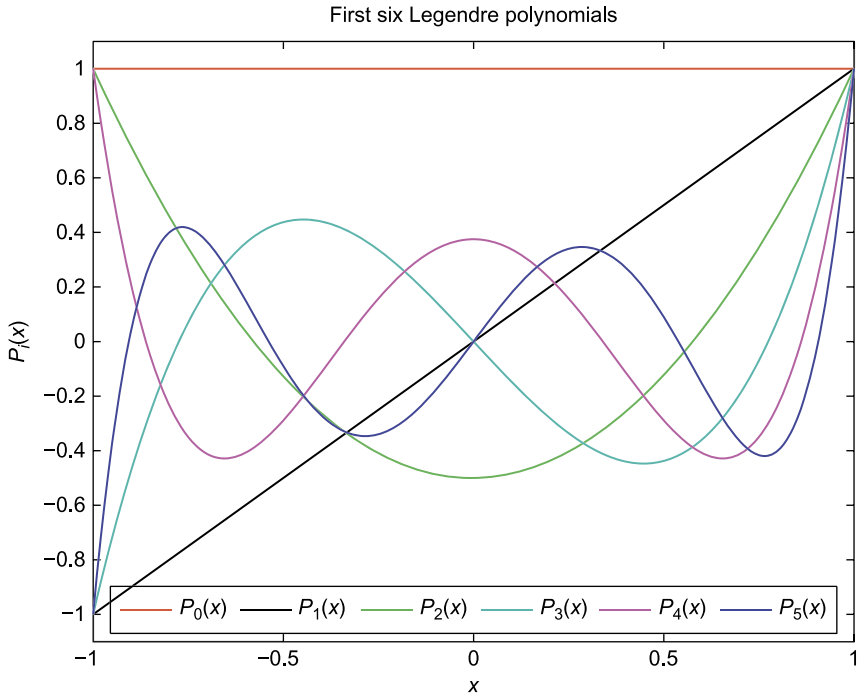


FIGURE 12.21

Plot of the first six Legendre polynomials.

12.5.4 Spherical Harmonics

The starting point for deriving the associated basis functions for spherical harmonics is to consider Laplace's equation in three dimensions on the sphere. Laplace's equation is equivalent to imposing the condition that the divergence of the gradient of a scalar field, f, is zero. Therefore, Laplace's equation in spherical coordinates is given by

$$\nabla^2 f = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial f}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial f}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2 f}{\partial\lambda^2} = 0, \tag{12.156}$$

where r is the radius in the outward direction, θ is the colatitude which is $\theta \in [0, \pi]$, and λ is the longitude which is $\lambda \in [0, 2\pi)$.

We start to solve (12.156) by assuming a separation of variables solution of the form

$$f(r, \theta, \lambda) = R(r)\, Y(\theta, \lambda), \tag{12.157}$$

and imposing the Laplace equation in (12.156), we obtain the following two differential equations;

$$\frac{1}{R}\frac{d}{dr}\left(r^2\frac{dR}{dr}\right) = \alpha, \tag{12.158a}$$

$$\frac{1}{Y}\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial Y}{\partial\theta}\right) + \frac{1}{Y}\frac{1}{Y}\frac{1}{\sin^2\theta}\frac{\partial^2 Y}{\partial\lambda^2} = -\alpha, \tag{12.158b}$$

where we have used the property from the Laplace equation that the sum of these two equations must equal zero, and as such they are equal to a constant but of opposite sign, and where we have made the assumptions that (1) f is not identically equal to zero, and by association; (2) neither R nor Y are equal to zero.

We now go a step further and assume that we can apply the separation of variable technique to the function Y to separate it into the product to two functions such that $Y(\theta, \lambda) = \Theta(\theta)\,\Lambda(\lambda)$, which then means that we can rewrite (12.158b) in terms of two differential equations given by

$$\frac{1}{\Lambda}\frac{d^2\Lambda}{d\lambda} = -m^2, \tag{12.159a}$$

$$\alpha\sin^2\theta + \frac{\sin\theta}{\Theta}\frac{d}{d\theta}\left(\sin\theta\frac{d\Theta}{d\theta}\right) = m^2, \tag{12.159b}$$

for some number m which happens to be complex constant, but because Λ must be a periodic function whose period evenly divides 2π, m is necessarily an integer, and Λ is a linear combination of the complex exponentials $e^{\pm im\lambda}$, that is to say that the solution to the ordinary differential equation in (12.159a) is $e^{\pm im\lambda}$.

For the ordinary differential equation in (12.159b), we have already shown that if we have a regular condition for Θ at the poles, $\theta = 0$ and $\theta = \pi$, then this makes (12.159b) a Sturm-Liouville problem, that we have already shown forces the parameter α to be equal to $\alpha = l(l+1)$ for some non-negative integer l such that $l \geq |m|$. Also if we use the change of variable, $\tau = \cos\theta$, then we now have the Legendre equation, whose solution is a multiple of the associated Legendre polynomial $P_l^m(\cos\theta)$. Given the expression we have for α, which, when substituted for the ordinary differential equations for r, yields the solution, $R(r) = Ar^l + Br^{-l-1}$; however, we require the solution to be regular throughout $\mathbb{R}^3$ that forces $B = 0$.

We recall that we assumed the solution has the special form $Y(\theta, \lambda) = \Theta(\theta)\,\Lambda(\lambda)$. For a given value of l, there are $2l+1$ independent solutions of this form, one for each integer m with $-l \leq m \leq l$. These solutions are referred to as **angular solutions** and are the product of trigonometric functions and the associated Legendre polynomials in the form of

$$Y_l^m(\theta, \lambda) = Ne^{\pm im\lambda} P_l^m(\cos\theta), \tag{12.160}$$

which satisfies

$$r^2\nabla^2 Y_l^m(\theta,\lambda) = -(l+1)\,Y_l^m(\theta,\lambda). \tag{12.161}$$

Here Y_l^m is called a **spherical harmonic function** of degree l and order m, P_l^m is an associated Legendre polynomial, and N is a normalization constant. For a fixed integer, l, every solution $Y(\theta,\lambda)$ of the eigenvalue problem

$$r^2\nabla^2 Y = -l(l+1)\,Y, \tag{12.162}$$

is a linear combination of Y_l^m.

12.5.5 Legendre Transforms

In the last section we introduced the continuous Fourier and Laplace transforms as mechanisms to simply certain operations so that the resulting differential or algebraic equations were simpler to solve than the original differential equations. Another such transform, and one that is used in spectral modeling in numerical weather predictions, is the **Legendre transform.**

We start by considering a convex function, where by convex it is meant that a real-valued function defined on an interval is convex if the line segment between any two points on the graph of the function lies above, or on, the graph. Given this requirement for $f(x)$, we define a new function $f^*(p)$ such that

$$f^*(p) \equiv \max_x (px - f(x)), \tag{12.163}$$

which is referred to as the **Legendre transform** of $f(x)$. If the function f is differentiable as well, then we can calculate the maximum through

$$0 = \frac{d}{dx}(px - f(x)) = p - \frac{df(x)}{dx},$$

whose solution depends on p, denoted as $x(p)$, and as such we have

$$\left.\frac{d(x)}{dx}\right|_{x=x(p)} = p, \tag{12.164}$$

whereupon substituting (12.164) into (12.163) yields

$$f^*(p) = px(p) - f(x(p)). \tag{12.165}$$

If we now consider the Legendre transform of $f^*(p)$, i.e., of (12.165), then we have

$$(f^*)^*(y) = \max_p \left(yp - f^*(p)\right). \tag{12.166}$$

If we now assume that the Legendre transform is differentiable with respect to p, then we have

$$\left.\frac{df^*(p)}{dp}\right|_{p=p(y)} = y. \tag{12.167}$$

However, if we now substitute all of the known expressions from above into (12.167), then we obtain

$$\frac{y^*(p)}{dp} = \frac{d}{dp}\left(px(p) - f(x(p))\right) = x(p) + p\frac{df(x)}{dp} - \left.\frac{df(x)}{dx}\right|_{x=x(p)} \frac{dx(p)}{dp} = x(p), \tag{12.168}$$

where we have used (12.164) to obtain the cancelation in (12.168). Given the result in (12.168), we now have that

$$f^{**}(y) = yp(y) - f^*(p(y)) = yp(y) - p(y)xp(y) + f(x(p(y))) = f(y). \tag{12.169}$$

The importance of (12.169) is that it tells us that the inverse of the Legendre transform is the Legendre transform itself, that is to say, the transform is its own inverse.

In numerical modeling we deal with multivariate problems, and so we require the transform to be extendable to this situation, and it is. The Legendre transform for the multivariate case is given by

$$f^*(p_1, p_2, \ldots, p_n) = \sum_{n=1}^{m} x_n p_n - f(x_1, x_2, \ldots, x_n), \text{ where } p_n = \frac{\partial f}{\partial x_n}. \tag{12.170}$$

Given all the theory that we have presented, we now move on to introduce the spectral methods for spherical coordinates.

12.5.6 Spectral Methods on the Sphere

The use of spectral methods on the sphere was first suggested in 1954 in [395]. As we have seen for the theory for spectral methods in Cartesian coordinates, we are able to represent a function, F, that is defined on the sphere as

$$F(\lambda, \theta) = \sum_{m=-M}^{M} \sum_{n=|m|}^{K} F_n^m Y_n^m(\lambda, \theta), \tag{12.171}$$

where m is referred to as the zonal wave number, n is the total wave number, and $n - |m|$ represents the effective meridional wave number. Given the expression in (12.171), it is possible to consider different truncations. Two of the more well-known and used truncations are the **triangular** and the **rhomboidal** truncation. These two truncations are defined as follows:

- If $K = M$, then the truncation is **triangular** and for some applications of this truncation if $m = 100$, then the model is said to be denoted $T100$.
- If $K = M + |m|$, then the truncation is a **rhomboidal** truncation.

To help illustrate why the two different truncations have these labeled we have presented the first five rows of both grids in Fig. 12.22.

Given that we are considering spectral approximation on the sphere, then it seems natural to use the spherical harmonic as the expansion functions, which implies

$$Y_n^m(\lambda\theta) = e^{im\lambda} P_n^m(\sin\theta), \tag{12.172}$$

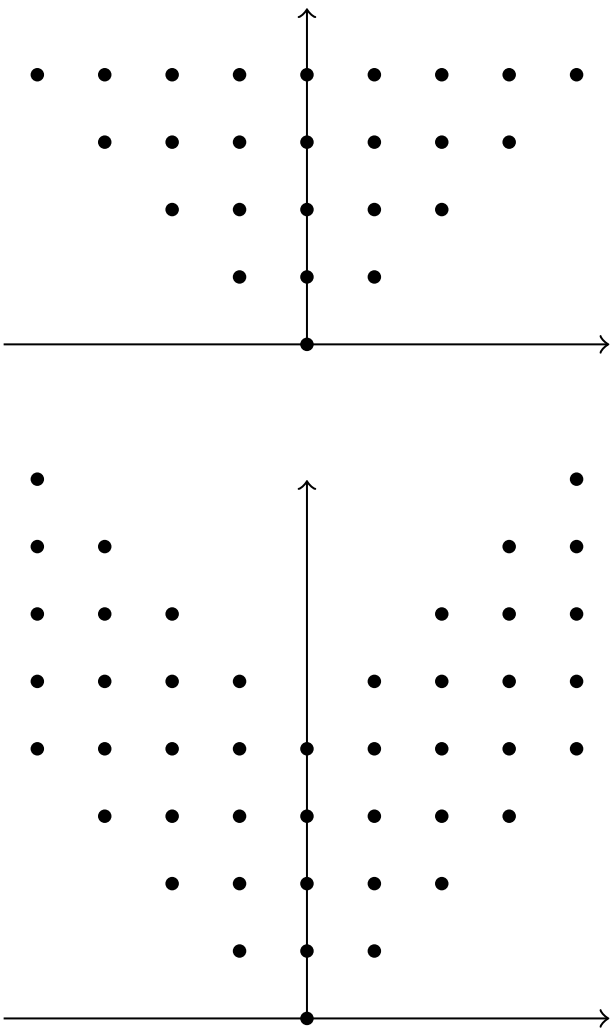


FIGURE 12.22

Triangular and rhomboidal truncations for $M = 4$.

where P_n^m are the associated Legendre polynomials and have the property

$$P_{-m}^n = P_m^n.$$

The spherical harmonics have the property

$$\nabla^2 Y_n^m = -\frac{n\,(n+1)}{a^2} Y_n^m, \tag{12.173}$$

where a is the Earth's radius.

We recall the property that the Legendre polynomials are orthogonal, and therefore satisfy

$$\frac{1}{2}\int_{-1}^{1} P_n^m(y)\, P_s^m(y)\, dy = \delta_{n,s}. \tag{12.174}$$

The space derivative of the spherical harmonic can be computed analytically as

$$\frac{\partial}{\partial\lambda} Y_n^m = imY_n^m, \tag{12.175}$$

and using the property of the Legendre polynomial, we have

$$\left(1-y^2\right)\frac{\partial}{\partial y} Y_n^m = -n\varepsilon_{n+1}^m Y_{n+1}^m + (n+1)\,\varepsilon_n^m Y_{n-1}^m, \text{ where } \varepsilon_n^m \equiv \sqrt{\frac{n^2-m^2}{4(n-1)}} \tag{12.176}$$

for $m \geq 0$. Another important property is that for $m > 0$, we have $Y_n^{-m} = \overline{Y_n^m}$.

Given the properties just described, it possible to calculate the space derivatives exactly, which then leaves a set of ordinary differential equations for the rate of change of the spherical harmonics coefficients, F_n^m.

However, as we saw with the Cartesian-based spectral theory, it is highly likely that we shall have nonlinear terms that we need to approximate, where two spherical harmonics interact to produce a third spherical harmonic. Unless the truncation is quite small, the calculations involved for this nonlinear interaction are very time consuming. As we saw in the Cartesian section, we can overcome this problem by introducing transformations. The three steps involved in this process are as follows:

(a) Starting in spectral space, the spectral coefficients are used to calculate the dependent variables on a latitude-longitude grid through an inverse spectral transform. If the regularly spaced longitudinal grid with at least $2M+1$ grid points and a specifically chosen latitude grid, about which we shall go into more details soon, then the transformations can be performed exactly.
(b) The nonlinear dynamics and physical process terms are calculated in real space.
(c) The nonlinear terms are transformed back to the spectral domain, which is referred to as the direct spectral transform.

To be able to perform the spectral transform, we introduce the Fourier coefficients as

$$F_n^m(y,t) = \frac{1}{2\pi}\int_0^{2\pi} F(\lambda,y,t)\, e^{-im\lambda} d\lambda = \sum_{n=|m|}^{N} F_n^m(t)\, P_n^m(y). \tag{12.177}$$

Next we multiply (12.171) by each of the spherical harmonics and use the orthogonality properties of both the Fourier basis functions and the Legendre polynomial, which leads to

$$F_n^m(t) = \frac{1}{4\pi}\int_0^1\int_0^{2\pi} F_n^m(\lambda,y,t)\, P_n^m(y)\, e^{-im\lambda} d\lambda dy. \tag{12.178}$$

The transform defined in (12.178) is referred to as the **direct spectral transform**.

To be able to calculate this transform, we first evaluate the integral with respect to λ. This is a Fourier transform and as such, if the data is at grid point in the longitudinal direction then this can be achieved through a fast Fourier, or discrete Fourier, transform. The transform in the longitudinal direction will be exact if the number of points in this direction is at least $2M + 1$, and equally spaced.

The integral in the latitudinal direction in the Fourier coefficients is approximated by a **Gaussian quadrature** formula, and it can be shown that this integral will be exact if the data points are at the points where the Legendre polynomials equal zero. This transform in the θ direction is the **Legendre transform**. We saw for each Legendre polynomial that the number of zeros of the polynomials is the same as the order of the polynomial. The points in the longitudinal direction that correspond to these zeros of the Legendre polynomial are referred to as the **Gaussian latitudes** and the number of the Gaussian latitude points for the solution to be exact must be $N_G \geq \frac{(2M+1)}{2}$. If we are multiplying two functions and we desire the approximation to be free of aliasing errors, then it can be shown that we require $N_G \geq \frac{3M+1}{2}$.

As we saw in the plot of the first six Legendre polynomials, the zeros of the polynomials were **not** equally spaced. The grid associated with the requirements just mentioned is referred to as the **Gaussian grid**. The grid associated with the single function when $N_G \geq \frac{2M+1}{2}$ is the **Gaussian linear grid**, while the grid associated with the product of two functions, if $N_G \geq \frac{3M+1}{2}$, is referred to as the **Gaussian quadratic grid**. An important feature of the Legendre polynomials which transfers to the grid points of the Gaussian grid is that due to the regularity condition of the solution to the Sturm-Liouville equation, there are no points at the poles.

We now provide a short algorithmic description of how to implement the spectral method on the sphere.

Step 1: Select a specific spectral truncation. Given this truncation, we identify the minimum number of grid points required in the longitudinal and latitudinal direction to avoid aliasing due to the quadratic nonlinearities.

Step 2: Identify the highest-degree Legendre polynomial required with the chosen spectral truncation, and then find the roots of this polynomial, which will be the Gaussian latitudes.

Step 3: Evaluate the horizontal derivatives in the spectral domain.

Step 4: Transform from spectral space to the Gaussian grid point space through the use of the inverse discrete Fourier transform, by applying the inverse fast Fourier transform techniques, and also the inverse Legendre transform.

Step 5: Evaluate the nonlinear terms in the approximation to the differential equation along with model physics on the Gaussian grid.

Step 6: Apply the fast Fourier transform and the Legendre transform to the grid point values to obtain the spectral coefficients.

Step 7: Apply a time-stepping scheme to advance the model and then repeat from Step 3 at the next time level.

12.6 Summary

In this chapter we have introduced some of theory that is required to be able to apply numerical approximations on the sphere, along with introducing the spectral methods which are different sets of

numerical approximations to different forms of ordinary and partial differential equations with spherical and Cartesian coordinates. We have introduced the spherical harmonic functions as the basis functions for the spectral methods on the sphere which are used operationally with differently configured Gaussian grids as well as total wavenumber. We have introduced the technique of the discrete Fourier transform, the fast Fourier transform, and their inverses, along with the Legendre transform and its inverse, which are used to convert between the Gaussian grids to perform nonlinear multiplications and then convert these values back into spectral space.

We have shown how to use finite difference methods on a regular latitudinal-longitudinal grid, along with the staggered and non-staggered grid formulations. We have also introduced the spherical differential operators, which now include the metric terms associated with the transform to that projections. We have presented some different spherical projections which are used not only in the numerical modeling part of data assimilation but also in how the observational data from different sources are stored.

We now move on to our last numerical model chapter, where we are going to develop the theory of tangent linear modeling along with the adjoints, which play an important role in four-dimensional variational data assimilation as well as in ensemble forecasting systems.

This page intentionally left blank

CHAPTER

Tangent Linear Modeling and Adjoints

13

Contents

In the derivation of what would form the basis of four-dimensional variational data assimilation, Lewis and Derber, [251], employed the use of a linearized model, and its adjoint, to evaluate the gradient of a penalty function to be able to obtain a set of initial conditions for a numerical model, given observed values at a set of different times, so that the model trajectory would fit better to the observations.

With the advent of incremental 4D VAR systems, it was shown that if we assume that we have what we believe to be a very good approximation to the true trajectory from a nonlinear numerical model, then we are only seeking a small change in the initial conditions to minimize a penalty function through a period of time. However, it would be impractical to run the full nonlinear model forward in time for such a small change, and as such we make the assumption that

$$\mathcal{M}(\mathbf{x}_b + \delta\mathbf{x}) \approx \mathcal{M}(\mathbf{x}_b) + \mathbf{M}\delta\mathbf{x}, \tag{13.1}$$

where $\mathcal{M}$ represents the full nonlinear numerical model and $\mathbf{M}$ is the linearized version of the model. We now go into more detail about the derivation of this linear model.

We shall go into more detail about the tangent linear model technique used in [251] and in incremental 4D VAR in Chapter 16, but for now we introduce the techniques that will be used in those schemes.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00017-6

13.1 Additive Tangent Linear and Adjoint Modeling Theory

In this section we shall introduce the additive tangent linear model and its adjoint. We shall also present testing procedures to verify the accuracy of the derivations, as well as the notion of differentiating the code. The reason why we refer to the derivation in this section as the additive tangent linear is due to the work by Fletcher and Jones [132], where it was necessary to adapt the tangent linear assumption. We shall go into more detail later in this chapter about this alternative form of tangent linear modeling, but we shall just clarify why we were making the distinction here.

13.1.1 Derivation of the Linearized Model

If we consider a general nonlinear initial value problem denoted by

$$y_i = \mathcal{M}(x_1, x_2, \ldots, x_N), \tag{13.2}$$

where N is the total number of grid points in the numerical grid, the x_js are the discrete model variables, for $j = 1, 2, \ldots, N$, and the y_is are the output from the numerical model, then what we may be seeking in various applications of data assimilation are answers to the questions of: how does $x\left(t^{n+1}\right)$ change with respect to $x_i\left(t^n\right)$? and when we are analyzing the output y_i, what features in x_j causes this? It is possible to attempt to quantify answers to these two questions by considering tangent linear models and adjoint models.

To derive the linearized model, we start by expressing the x_js as either a background, or reference state, and a perturbation such that

$$x_j = \overline{x}_j + \delta x_j. \tag{13.3}$$

We are trying to ascertain how the nonlinear model is affected by the perturbation to the state, x_j, which can be approximated through considering the difference between the outputs from $y(\overline{x} + \delta x)$ and $y(\overline{x})$. We have plotted this situation in Fig. 13.1, where we see that we can approximate the change in the

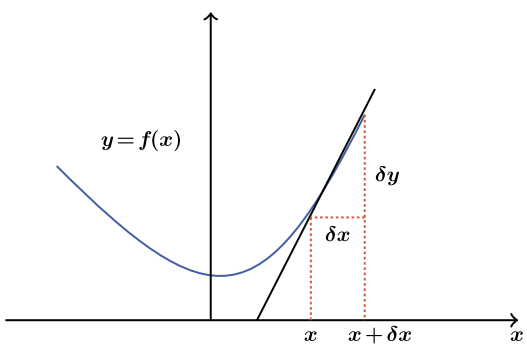


FIGURE 13.1

Schematic of the additive tangent linear approximation to the function $f(x)$.

output, δy, through a tangent approximation. Mathematically we can express the gradient as

$$\frac{dy}{dx} \approx \frac{\delta y_i}{\delta x_i} \Rightarrow \delta y_i \approx \left(y_i\left(\overline{x}+\delta x\right) - y_i\left(\overline{x}\right)\right)\delta x_i . \tag{13.4}$$

We now substitute the nonlinear model for y in (13.4), and expand the nonlinear model about the reference state to obtain

$$\begin{aligned}
\delta y_i &= y_i\left(\overline{x}+\delta x\right) - y_i\left(\overline{x}\right) = \mathcal{M}\left(\overline{x}_1+\delta x_1, \overline{x}_2+\delta x_2, \ldots, \overline{x}_N+\delta x_N\right) - \mathcal{M}\left(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_N\right), \\
&= \mathcal{M}\left(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_N\right) + \frac{\partial \overline{y}_i}{\partial x_1}\delta x_1 + \frac{\partial \overline{y}_i}{\partial x_2}\delta x_2 + \cdots - \mathcal{M}\left(\overline{x}_1, \overline{x}_2, \ldots, \overline{x}_N\right), \\
\delta y_i &= \sum_j^N \left(\overline{\frac{\partial y_i}{\partial x_j}}\right)\delta x_j .
\end{aligned} \tag{13.5}$$

The expression in (13.5) is the **tangent linear model**, which is quite often abbreviated to TLM, and it gives an approximation for the growth of the perturbations, δx_j, $j = 1, 2, \ldots, N$. The tangent linear model enables us to address the question of how y_i changes with respect to x_j, for $j = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, M$, where M is the number of time steps the numerical model has taken.

13.1.2 Adjoints

We now consider how to address the second question of linking the behavior in y_i to x_j. We start by introducing a scalar measure of the outputs

$$J = J\left(\boldsymbol{y}\left(\boldsymbol{x}\right)\right). \tag{13.6}$$

We next expand (13.6) through a Taylor series, which leads to a change in the measure, δJ, as

$$\delta J = \sum_{i=1}^{M} \frac{\partial \overline{J}}{\partial y_i}\delta y_i , \tag{13.7}$$

but we can also consider variations with respect to the x_js, which results in

$$\delta J = \sum_{j=1}^{N} \frac{\partial \overline{J}}{\partial x_j}\delta x_j , \tag{13.8}$$

where (13.7) and (13.8) are **equal when both functions are linear**.

It is (13.8) that is of interest here, as it depends on the perturbation inputs. Given how we defined J as a function of a function, it is possible to differentiate through the chain rule to obtain estimates for $\frac{\partial \overline{J}}{\partial x_j}$ as

$$\frac{\partial \overline{J}}{\partial x_j} = \sum_{i=1}^{M} \left(\overline{\frac{\partial y_i}{\partial x_j}}\right)\frac{\partial \overline{J}}{\partial y_i}. \tag{13.9}$$

An important, and arguably very powerful property of (13.9), is that it enables us to integrate the future time gradient, or sensitivities, **backwards** to the initial time.

The first feature to notice here is that the tangent linear model sums the terms $\frac{\partial y_i}{\partial x_j}$ over the js, while the summation in (13.9) is over the is. To help keep track of the different terms here, we shall now use matrix-vector notation. First let the matrix $\mathbf{M}$ be defined as

$$\mathbf{M}_{i,j} \equiv \frac{\partial y_i}{\partial x_j}, \tag{13.10}$$

then for the tangent linear model we see that we are summing over the columns of $\mathbf{M}$, while for the model given by (13.9) we are summing over the rows of $\mathbf{M}$. From the rules of matrix-vector multiplication and introducing the vectors $\delta \boldsymbol{y}$ and $\delta \boldsymbol{x}$, we can write (13.5) as

$$\delta \boldsymbol{y} = \mathbf{M} \delta \boldsymbol{x}. \tag{13.11}$$

We can then also write (13.9) as

$$\frac{\partial J}{\partial \boldsymbol{x}} = \mathbf{M}^T \frac{\partial J}{\partial \boldsymbol{y}}. \tag{13.12}$$

The matrix $\mathbf{M}$ as defined in (13.10) is referred to as either the **resolvent** of the tangent linear model, or as the **Jacobian** of the nonlinear model. Given the definition for the resolvent, it is possible to write the perturbation to the nonlinear model as

$$\mathcal{M}(\overline{\boldsymbol{x}} + \delta \boldsymbol{x}) = \mathcal{M}(\overline{\boldsymbol{x}}) + \mathbf{M} \delta \boldsymbol{x}, \tag{13.13}$$

which is equivalent to the expression in the introduction to this chapter. As an aside, depending on the papers or textbooks you read, the resolvent is also denoted by $\mathbf{L}$.

The model that maps sensitivities backwards in referred to as the **adjoint model** as it uses the adjoint of the tangent linear model.

There are many different ways to derive the adjoint of a tangent linear model. The first approach is to derive the tangent linear model of either the discrete or continuous model; the second approach is referred to as **differentiating the code**.

Example 13.1. *Derive the tangent linear model and its adjoint for the one-dimensional tracer advection problem, for the tracer $q(x, u, t)$, that is given by*

$$\frac{\partial q}{\partial t} + u \frac{\partial q}{\partial x} = 0. \tag{13.14}$$

The first step in deriving the tangent linear and adjoint models is to introduce perturbations for q and u as $\overline{q} + \delta q$ and $\overline{u} + \delta u$, and then substitute these perturbations into (13.14); forming the continuous tangent linear equations, we have

$$\frac{\partial \delta q}{\partial t} + \delta u \frac{\overline{q}}{\partial x} + \overline{u} \frac{\partial \delta q}{\partial x} + \delta u \frac{\partial \delta q}{\partial x} = 0. \tag{13.15}$$

The second-order term in δ is ignored in the tangent linear approximation. If we now discretize (13.15) with a forward upwind scheme in both space and time, then we have

$$\delta q_j^{n+1} = \delta q_j^n - \Delta t \left(\delta u_j^n \frac{\overline{q}_{j+1}^n - \overline{q}_j^n}{\Delta x} - \overline{u}_j^n \frac{\delta q_{j+1}^n - \delta q_j^n}{\Delta x} \right) = 0. \tag{13.16}$$

Before we progress any further we introduce the notion of the **active** variables for the derivation of the adjoint. The definition of the active variables comes from Kalnay [208], where active variables are those whose values that are used to **modify** a set, or all, of the active variables. An important property that needs to be maintained in the coding of the adjoint is that if an active variable is not modified, that is to say that there is no equation for that variable, but it appears in the equation that does modify a different active variable, then we need to ensure that the unmodified variable is not changed. One approach to ensure that unmodified variables remains unchanged is achieved by including the condition $q_j^n = q_j^n$.

Returning to the advection example, we can identify that we have four active variables in (13.16); these are δq_j^{n+1}, δq_j^n, δu_j^n, and δq_{j+1}^n, but only one of them is modified, namely q_j^{n+1} therefore we need to ensure that the remaining three are not modified in the tangent linear model. Thus, we can write (13.16) as a matrix-vector multiplication given by

$$\begin{pmatrix} \delta q_{j+1}^n \\ \delta q_j^n \\ \delta u_j^n \\ \delta q_j^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1+b_j^n & -b_j^n & a_j^n & 0 \end{pmatrix} \begin{pmatrix} \delta q_{j+1}^n \\ \delta q_j^n \\ \delta u_j^n \\ \delta q_j^{n+1} \end{pmatrix}, \tag{13.17}$$

where

$$a_j^n = -\frac{\Delta t}{\Delta x}\left(\overline{q}_{j+1}^n - \overline{q}_j\right), \quad b_j^n \equiv \overline{u}_j^n \frac{\Delta t}{\Delta x}.$$

Introducing the adjoint variables $\begin{pmatrix} \widehat{\delta q_{j+1}^n} \\ \widehat{\delta q_j^n} \\ \widehat{\delta u_j^n} \\ \widehat{\delta q_j^{n+1}} \end{pmatrix}$, we obtain the adjoint equations by taking the transpose of the matrix in (13.17) and multiplying out the matrix-vector multiplications, which results in

$$\begin{aligned} \widehat{\delta q_{j+1}^n} &= \widehat{\delta q_{j+1}^n} - b_j^n \widehat{\delta q_{j+1}^{n+!}}, \\ \widehat{\delta q_j^n} &= \widehat{\delta q_j^n} + \left(1 + b_j^n\right) \widehat{\delta q_{j+1}^{n+1}}, \\ \widehat{\delta u_j^n} &= \widehat{\delta u_j^n} + a_j^n \widehat{\delta q_{j+1}^{n+1}}, \\ \widehat{\delta q_{j+1}^{n+1}} &= 0. \end{aligned}$$

Exercise 13.2. *Derive the tangent linear model and the adjoint for a centered-time, centered-space scheme applied to* (13.14).

13.1.3 Differentiating the Code to Derive the Adjoint

In the derivations of the adjoints that we have presented so far, we have either used the continuous forms to derive the tangent linear equations and then derived the continuous form of the adjoint equations, or we have taken the discrete equations for the linear advection equations in a matrix form, and then

formed the transpose of that matrix to form the adjoint model. It is often the case that we have a set of computer codes of the tangent linear model and it is shown in [316] that it is possible to *differentiate the computer code* rather than having to derive the equations themselves. We now present a summary of the appendix from Navon et al. [316], who show the equivalency of differentiation of the code to the transpose of the tangent linear matrix.

If we consider a linear model which consists of a DO, or a FOR, loop depending on the computing language you are using, then we have

$$\begin{aligned}&\text{DO } I = 1, N-1,\\ &X(I) = aY(I+1),\\ &\text{END DO},\end{aligned}$$

where X and Y are vectors. This DO loop is equivalent to the following algebraic matrix equations:

$$\begin{pmatrix} X(1)\\ X(2)\\ \vdots\\ \vdots\\ \vdots\\ X(N-1)\end{pmatrix} = \begin{pmatrix} 0 & a & 0 & \cdots & 0 & 0\\ 0 & 0 & a & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & 0 & a\end{pmatrix}\begin{pmatrix} Y(1)\\ Y(2)\\ \vdots\\ \vdots\\ \vdots\\ Y(N)\end{pmatrix}, \tag{13.18}$$

if the $Y(I)$s are not reused outside of the loop, or if the variables are used outside of the loop, then the matrix equations become

$$\begin{pmatrix} Y(1)\\ Y(2)\\ \vdots\\ \vdots\\ \vdots\\ Y(N)\\ X(1)\\ X(2)\\ \vdots\\ \vdots\\ \vdots\\ X(N)\end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0\\ 0 & 1 & 0 & \cdots & 0 & 0\\ 0 & 0 & 1 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & 0 & 1\\ 0 & a & 0 & \cdots & 0 & 0\\ 0 & 0 & a & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots\\ 0 & 0 & 0 & \cdots & 0 & a\\ 0 & 0 & 0 & \cdots & 0 & 0\end{pmatrix}\begin{pmatrix} Y(1)\\ Y(2)\\ \vdots\\ \vdots\\ \vdots\\ Y(N)\end{pmatrix}. \tag{13.19}$$

We know that the adjoints of (13.18) and (13.19) are given by the transpose of the matrices in these equations, which yields

$$\begin{pmatrix} \widehat{Y(1)} \\ \widehat{Y(2)} \\ \vdots \\ \vdots \\ \vdots \\ \widehat{Y(N)} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ a & 0 & 0 & \cdots & 0 & 0 \\ 0 & a0 & \cdots & 0 & 0 & \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \begin{pmatrix} \widehat{X(1)} \\ \widehat{X(2)} \\ \vdots \\ \vdots \\ \vdots \\ \widehat{X(N-1)} \end{pmatrix}, \tag{13.20}$$

for (13.18) and

$$\begin{pmatrix} \widehat{Y(1)} \\ \widehat{Y(2)} \\ \vdots \\ \vdots \\ \vdots \\ \widehat{Y(N)} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & a & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 & a & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & 0 & \cdots & a & 0 \end{pmatrix} \begin{pmatrix} \widehat{Y(1)} \\ \vdots \\ \widehat{Y(N)} \\ \widehat{X(1)} \\ \vdots \\ \widehat{X(N)} \end{pmatrix}, \tag{13.21}$$

for (13.19). Therefore, the adjoint of the do loop will be in one of two forms, either

$$\begin{aligned} &\text{DO } I = 1, N-1 \\ &YHAT\,(I+1) = aXHAT\,(I) \\ &\text{END DO,} \end{aligned}$$

or

$$\begin{aligned} &\text{DO } I = 1, N-1 \\ &YHAT\,(I+1) = YHAT\,(I+1) + aXHAT\,(I) \\ &\text{END DO.} \end{aligned}$$

The general rules for coding adjoints are:

1. Make a copy of the original tangent linear program.
2. Remove all of the lines in the program that are not acting upon the perturbations.
3. Reverse the order of **all** the operations, including the loops.
4. For each line inside of the loop, carry out the following:
   - **(a)** For each perturbation that occurs on the right-hand side of the equal sign in the tangent linear model, create a new line with that perturbation now being the left-hand side variable.
   - **(b)** Set these perturbations equal to themselves and then with a +.
   - **(c)** Take the original coefficient that was multiplying the perturbation, multiply it by the old left-hand side tangent linear variable, and then add it on to the perturbation.
5. Set the old left-hand side variable to zero.

We shall go through an example of how to derive the adjoint from the code for the Lorenz 1963 model later in this chapter.

13.1.4 Test of the Tangent Linear and Adjoints Models

There are a series of tests that can be applied to verify if the derivation and the coding of the tangent linear and adjoint models are correct. If we consider the tangent linear model first, then we have shown that if we have a nonlinear $\mathcal{M}(\boldsymbol{x})$, and $\mathbf{M}$ is the tangent linear model, then for small perturbations $\delta\boldsymbol{x}$ we have

$$\mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x})\approx\mathbf{M}\gamma\delta\boldsymbol{x}.$$

If we define the relative error as

$$E_R=\frac{\mathcal{M}(\boldsymbol{x}+\gamma\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x})}{\mathbf{M}\gamma\delta\boldsymbol{x}}, \tag{13.22}$$

and if the tangent linear approximation is a good approximation, then as $\gamma\rightarrow 0$ we should have that the relative error also tends to zero.

If we now consider the adjoint model, then we have the following identity for the tangent linear and adjoint model

$$\langle\mathbf{M}\delta\boldsymbol{x},\mathbf{M}\delta\boldsymbol{x}\rangle=\left\langle\delta\boldsymbol{x},\mathbf{M}^T\mathbf{M}\right\rangle, \tag{13.23}$$

for an inner product, $\langle\cdot,\cdot\rangle$, and any $\delta\boldsymbol{x}$. This is a mechanism to test the accuracy of the adjoint calculation.

The third measure of accuracy is through the **gradient test**. If we recall that we have a functional associated with the adjoint given by J, then the gradient of J is ∇J, and it is possible to check that the gradient of the functional has been coded correctly through the identity

$$\Psi(\alpha)=\frac{J(\boldsymbol{x}+\alpha\boldsymbol{h})-J(\boldsymbol{x})}{\alpha\boldsymbol{h}^T\nabla J(\boldsymbol{x})}=1+O(\alpha), \tag{13.24}$$

where $\boldsymbol{h}$ is a vector of unit length which is often taken to be $\dfrac{\nabla J(\boldsymbol{x})}{\|\nabla J(\boldsymbol{x})\|_2}$. Therefore, for some values of α away from the accuracy of the machine precision, if the gradient is accurate then we would expect the function $\Psi(\alpha)$ to be approximately 1.

13.2 Multiplicative Tangent Linear and Adjoint Modeling Theory

In the derivation of the additive tangent linear model, we saw that the approximation to obtain the model came about through considering the approximation to the gradient of the output function as the difference between $y(x+\delta x)$ and $y(x)$ divided by δx. However, in the development of lognormal-based incremental 3D and 4D VAR in Fletcher and Jones [132], it became apparent that a multiplicative version of tangent linear modeling was required.

If we now consider a **multiplicative increment**, Δx, which would imply a geometric behavior to the function, then we wish to determine the effect that this multiplicative increment has on the output of a nonlinear model. The motivation for the multiplicative, or geometric, tangent linear modeling is the same as for the additive tangent linear model, but now we have that, we wish to investigate the sensitivity to $y = \mathcal{M}(x\Delta x)$. If we just think about what we do in the tangent/gradient approximation, then we recall that we move along the x axis by adding a small change, well in [132] the authors state that we could also move from x_1 to a x_2 by **multiplying** x_1 by an increment that is close to 1, which could also move from x_1 to x_2. Given these values we could determine, as we do in the additive tangent linear case, the difference to the gradients as

$$\frac{dy}{dx} \approx \frac{\Delta y}{x_1 \Delta x - x_1} \Rightarrow \Delta y = (\mathcal{M}(x_1 \Delta x) - \mathcal{M}(x_1))(x_1(\Delta x - 1)), \tag{13.25}$$

where $\Delta x \approx 1$, so that as $\Delta x \to 1$ then $\Delta y \to 0$. We have drawn a diagram of the geometric tangent linear approximation in Fig. 13.2 to illustrate how this approximation is formed.

Tangent linear approximations play a vital role in incremental variational data assimilation; while the additive approach is compatible with additive-based distributions for the errors, it is not the case for geometric-based distributions. In [132] the authors took the approximation above and were able to prove that the linearization that was required for the geometric-based schemes were possible with the geometric formulations. Below are the theorems and the proof of these vital tools for geometric-based incremental variational data assimilation.

Theorem 13.3. *For continuous functions f and g and a multiplicative increment, $\Delta x \approx 1$, the first-order approximations of* $\ln f(x\Delta x)$ *and* $\ln f(g(x\Delta x))$ *are*

$$\ln f(x\Delta x) \approx \ln f(x) + \frac{1}{f(x)} \frac{df(x)}{dx} x(\Delta x - 1), \tag{13.26}$$

$$\ln f(g(x\Delta x)) \approx \ln f(g(x)) + \frac{1}{f(g(x))} \frac{df(g(x))}{dg(x)} \frac{dg(x)}{dx} x(\Delta x - 1). \tag{13.27}$$

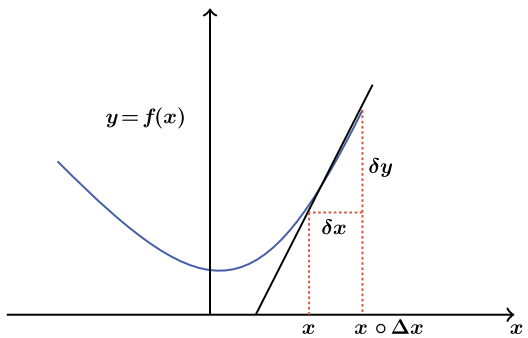


FIGURE 13.2

Schematic of the geometric tangent linear approximation to the function $f(x)$.

Proof. Consider the graph in Fig. 13.2, where $x_2 = x_1 \Delta x$ and $y_2 = f(x_1 \Delta x)$. Let $\delta x = x_2 - x_1$ and $\delta y = y_2 - y_1$ then $x_2 = x_1 + \delta x$ and $y_2 = y_1 + \delta y$. The definition of the gradient states that

$$\lim_{\delta x \to 0} \frac{\delta y}{\delta x} = \frac{dy}{dx}. \tag{13.28}$$

Expanding (13.28) with respect to the definition for δy above gives us

$$\lim_{\delta x \to 0} \frac{y_2 - y_1}{\delta x} = \frac{dy}{dx}. \tag{13.29}$$

Substituting for y_2 and y_1, we have

$$\lim_{\delta x \to 0} \frac{f(x_2) - f(x_1)}{\delta x} = \frac{dy}{dx}. \tag{13.30}$$

Now substituting the definition of x_2 in terms of x_1 and realizing that $\delta x = x_2 - x_1 \equiv x_1 \Delta x - x_1 = x_1(\Delta x - 1)$, we see that for the definition of the gradient to hold as $\delta x \to 0$, then $\Delta x \to 1$. This first step proves that the gradient can be approximated using a multiplicative increment. This enables the standard proofs of $\ln x$ and the chain rule to be applied. Therefore,

$$\begin{aligned} \lim_{\Delta x \to 1} \frac{\ln f(x \Delta x) - \ln f(x)}{x(\Delta x - 1)} &= \frac{1}{f(x)} \frac{df(x)}{dx}, \\ \Rightarrow \ln f(x \Delta x) &\approx \ln f(x) + \frac{1}{f(x)} \frac{df(x)}{dx} x(\Delta x - 1), \end{aligned} \tag{13.31}$$

and

$$\begin{aligned} \lim_{\Delta x \to 1} \frac{\ln f(g(x \Delta x)) - \ln f(g(x))}{x(\Delta x - 1)} &= \frac{1}{f(g(x))} \frac{df(g(x))}{dg(x)} \frac{dg(x)}{dx}, \\ \Rightarrow \ln f(g(x \Delta x)) &\approx \ln f(g(x)) + \frac{1}{f(g(x))} \frac{df(g(x))}{dg(x)} \frac{dg(x)}{dx} x(\Delta x - 1). \end{aligned} \tag{13.32}$$

Therefore, (13.31) and (13.32) prove (13.26) and (13.27) for $\Delta x \approx 1$.

13.3 Examples of Adjoint Derivations

In this section we shall apply the two different techniques for calculating the adjoint equations. The first is a direct calculation of the tangent linear equations matrix, taking the adjoint and then deriving the associated adjoint equation. The second approach is the differentiation of the code technique. The first model that we shall consider is the famous Lorenz 63 model.

13.3.1 Lorenz 63 Model

In 1963, Professor Edward Lorenz published a paper that changed our understanding of the sensitivity of numerical approximations of the atmosphere [270]. The derivation of the model is a simplification of the model derived by Saltzman in 1962 [380], to study finite-amplitude convection. The starting point of the derivation is from a paper by Rayleigh in 1916 [362], which is a study of flow occurring in a layer

of fluid of uniform depth H, when the temperature difference between the upper and lower surfaces is maintained at a constant value which he denotes by ΔT. It is stated in [270] that this type of system possesses a steady state solution, where there is no motion, and the temperature varies linearly with depth. Therefore, if the solution is unstable, then convection should develop.

If we now consider the case where all motions are parallel to the x-z plane, and there is no variation in the direction of the y-axis, then from [380] the governing equations of the motion are written as

$$\frac{\partial}{\partial t}\nabla^2\psi = -\frac{\partial\left(\psi, \nabla^2\psi\right)}{\partial\left(x, z\right)} + \nu\nabla^4\psi + g\alpha\frac{\partial\theta}{\partial x}, \tag{13.33a}$$

$$\frac{\partial}{\partial t}\theta = -\frac{\partial\left(\psi, \theta\right)}{\partial\left(x, z\right)} + \frac{\Delta T}{H}\frac{\partial\psi}{\partial x} + \kappa\nabla^2\theta, \tag{13.33b}$$

where ψ is a stream function for the two-dimensional motion, θ is the departure of the temperature from the value for the state of no convection, and the constants g, α, ν, κ denote the acceleration of gravity, the coefficient of thermal expansion, the kinematic viscosity, and the thermal conductivity, respectively. For this problem to be tractable, we have that if both the upper and lower boundaries are taken to be free, then ψ and $\nabla^2\psi$ vanish at both boundaries.

In [362] it is shown that the fields of motion of the form

$$\psi = \psi_0 \sin\left(\frac{\pi a x}{H}\right)\sin\left(\frac{\pi z}{H}\right), \tag{13.34a}$$

$$\theta = \theta_0 \cos\left(\frac{\pi a x}{H}\right)\sin\left(\frac{\pi z}{H}\right), \tag{13.34b}$$

would develop if the quantity

$$R_a = \frac{g\alpha H^3 \Delta T}{\nu\kappa}, \tag{13.35}$$

which is called the **Rayleigh number**, exceeded a critical value

$$R_c = \frac{\pi^4\left(1 + a^2\right)^3}{a^2}.$$

The minimum value of R_a is $\frac{27\pi^4}{4}$, which occurs at $a^2 = \frac{1}{2}$.

Given this formulation, Saltzman derived a set of ordinary differential equations through expanding ψ and θ by a double Fourier series in x and z, where the coefficients were functions of t only and substituted these Fourier series into (13.34a) and (13.34b). The right-hand side of these equations were arranged in double Fourier series form, by replacing products of trigonometric functions of either x or z, by sums of trigonometric functions and then equating coefficients of similar functions of x and z. The resulting infinite system was reduced to a finite system through omitting references to but a specific set of functions of t.

The next step in the derivation in [380] was to apply a numerical integration which resulted in time-dependent solutions. In all but three cases the dependent variables tended to zero; where these three remaining variables underwent irregular non-periodic fluctuations which are what caught Lorenz's attention.

Lorenz in [270] stated that the same solutions that Saltzman found could have been obtained if the series had been truncated to include a total of three terms, and as such we would obtain

$$\frac{a}{(1+a^2)\kappa}\psi = X\sqrt{2}\sin\left(\frac{\pi a x}{H}\right)\sin\left(\frac{\pi z}{H}\right), \tag{13.36a}$$

$$\frac{\pi R_a}{R_c \Delta T}\theta = Y\sqrt{2}\cos\left(\frac{\pi a x}{H}\right)\sin\left(\frac{\pi z}{H}\right) - Z\sin\left(\frac{2\pi z}{H}\right), \tag{13.36b}$$

where X, Y, and Z are functions of time only. When (13.36a) and (13.36b) are substituted in (13.34a) and (13.34b), and when the trigonometric terms other than those in (13.36a) and (13.36b) are omitted, we obtain the famous Lorenz 1963 model equations

$$\dot{x} = -\sigma x + \sigma y, \tag{13.37a}$$

$$\dot{y} = -xz + \rho x - y, \tag{13.37b}$$

$$\dot{z} = x + y - \beta z, \tag{13.37c}$$

where the superscript dot refers to the derivative with respect to a dimensionless time, $\tau = \frac{\pi(1+a^2)\kappa t}{H^2}$, $\sigma = \frac{\nu}{\kappa}$ is the **Prandtl number**, $\rho = \frac{R_a}{R_c}$ and $\beta \frac{4}{(1+a^2)}$. Therefore, the equations that make up the Lorenz model are those that model convection.

The physical meaning of the three variables are stated in [270] where x is proportional to the intensity of the convective motion, while y is proportional to the temperature difference between ascending and descending currents, similar signs of x and y denoting that warm fluid is rising and cold fluid is descending. The variable z is proportional to the distortion of the vertical profile from linearity.

To illustrate the sensitivity of the model to the parameters, we have plotted the solutions of the Lorenz model which has been discretized by the modified Euler numerical scheme. We shall go into more detail about this approximation soon, as this is the scheme that we shall be deriving the adjoint for, with the same initial conditions but with different values for the parameters. Our *standard* model uses the values $\sigma = 10$, $\rho = 28$, and $\beta = \frac{8}{3}$ for the three parameters which generate a chaotic solution. We start the model from $x\,(0) = 5.4$, $y\,(0) = -5.44$, and $z\,(0) = 22.5$, and we run the model out to 3000 time steps.

In the plots in Fig. 13.3 we have kept ρ and β at the standard model's values but investigate the values of 15, 12.5, 10, 7.5, 5, and 2.5 for σ. We can see that for the larger values of σ the trajectory transitions between what are called *attractors* of the model, where we are considering $x - z$ plots, at different frequencies. However, when we use $\sigma = 5$ then we appear to have a stable solution which means that it remains on one attractor. When $\sigma = 2.5$, we see that the solution collapses to the stable solution faster than $\sigma = 5$.

In Fig. 13.4 we have kept σ and β at the standard model's value and are now varying ρ where we use the values 48, 38, 28, 18, 8, and 0. We see quite a different appearance to the chaotic structure when we vary the ρ parameter compared to varying the σ parameter. However, we obtain the stable solution when we set $\rho = 18$, but when we take values below that we see that the solution is tending to a stable point faster and when we set $\rho = 0$ we tend to a zero solution.

Finally, in Fig. 13.5 we have varied the β parameter with the values $\frac{17}{3}$, $\frac{14}{3}$, $\frac{11}{3}$, $\frac{8}{3}$, $\frac{5}{3}$, and $\frac{2}{3}$. We see for the larger values of the β parameter that the model remains on one of the attractors; when we have the standard model we transition between the two attractors. When we reduce the β parameter below the standard model's values we also see a transition between the two attractors but less frequently, and when we have $\beta = \frac{2}{3}$ we have a stable solution but one that appears to transition between the two attractors quite frequently.

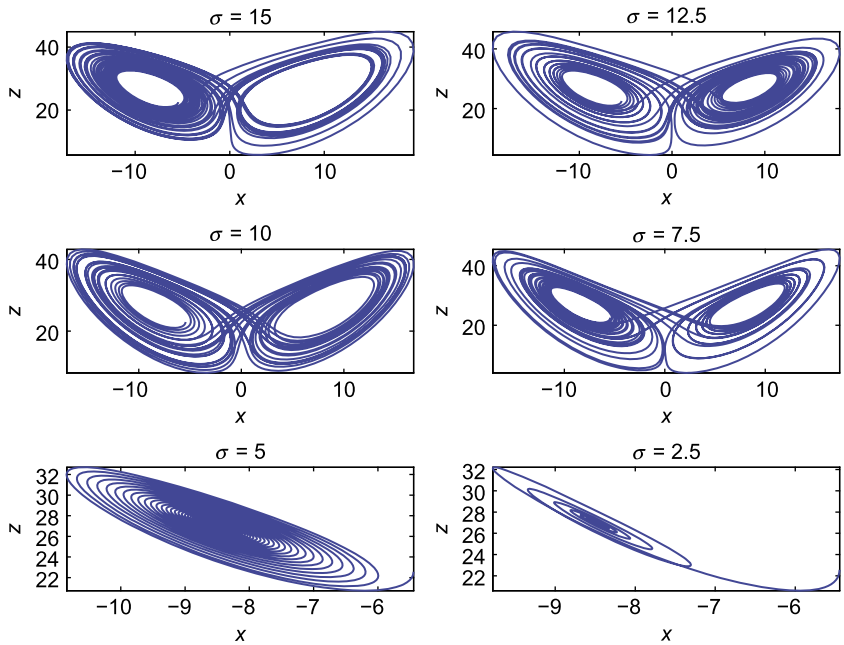


FIGURE 13.3

Plot of the sensitivity of the Lorenz 63 model to the choice of the σ parameter.

The reason why the Lorenz 63 model is a good model to test data assimilation development is because of the incredible sensitivity the model has to the initial conditions. The discovery of *chaos* by Lorenz came about due to his need to rerun an experiment from a set of points further along in the simulation. He truncated the initial conditions from the output from six figures of accuracy to three, but the result was a completely different answer to the one he already had. This sensitivity to the initial conditions is a problem for people using data assimilation, because we are nearly always seeking the best set of initial conditions, or a set of model parameters, but due to the sensitivity of the initial conditions, and the model to the values of the parameters as we have presented as well, for certain models then the outcome could be completely different from the true solution.

In Fig. 13.6 we have plotted the standard model with the same initial conditions from the parameter sensitivity work but now we have reduced the initial conditions by one decimal place for each plot and have subtracted the *true* state to present a form of *error* for each set of initial conditions. We see that when we are only out by one decimal point from the true state, then the two solutions appear quite similar to over 900 time steps, but after that the solutions become quite different. When we are different by two decimal points in the initial condition, the solutions start to diverge from each other at an earlier time. This difference between the two solutions when we are differing in the initial conditions at three decimal places is quite different to the true state almost straightaway. Finally, when only the integer component matches we see that the solutions differ from the start, which indicates that the model is very sensitive to its initial conditions.

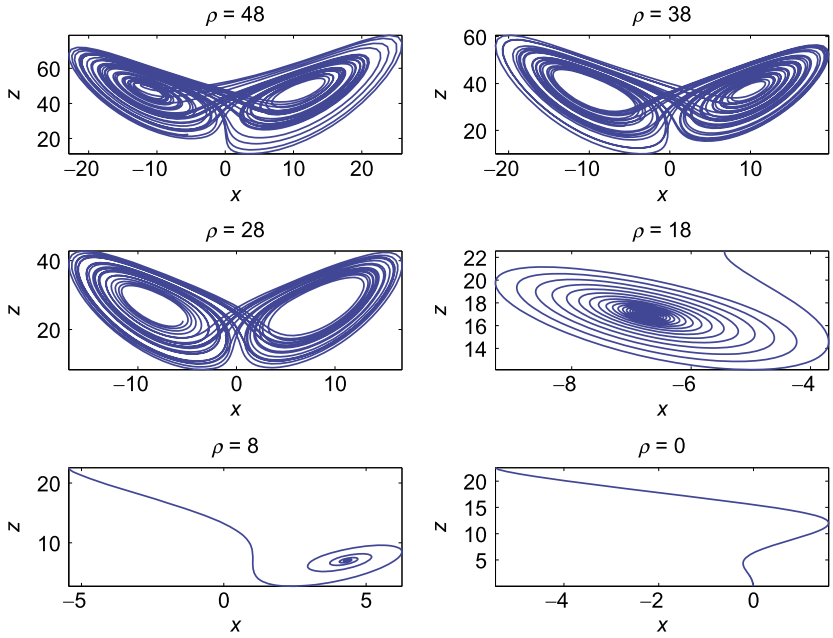


FIGURE 13.4

Plot of the sensitivity of the Lorenz 63 model to the choice of the ρ parameter.

We now consider how to build the tangent linear and adjoint of the Lorenz 63 model that has been discretized through the modified Euler scheme.

Modified Euler approximation to Lorenz 63 model

As we showed earlier, the modified Euler scheme is the same as the second-order Runge-Kutta scheme. The equations for the modified Euler scheme are given by

$$K_1(x_n) = h\sigma(y_n - x_n), \tag{13.38a}$$
$$K_1(y_n) = h(\rho x_n - y_n - x_n z_n), \tag{13.38b}$$
$$K_1(z_n) = h(x_x y_n - \beta z_n), \tag{13.38c}$$
$$K_2(x_n) = h(\sigma(y_n + K_1(y_n) - x_n - K_1(x_n))), \tag{13.38d}$$
$$K_2(y_n) = h(\rho(x_n + K_1(x_n)) - y_n - K_1(y_n) - (x_n + K_1(x_n))(z_n + K_1(z_n))), \tag{13.38e}$$
$$K_2(z_n) = h((x_n + K_1(x_n))(y_n + K_1(y_n)) - \beta(z_n + K_1(z_n))). \tag{13.38f}$$

The final approximation to the Lorenz equations is given by

$$x_{n+1} = x_n + \frac{K_1(x_n) + K_2(x_n)}{2}, \tag{13.39a}$$
$$y_{n+1} = y_n + \frac{K_1(y_n) + K_2(y_n)}{2}, \tag{13.39b}$$

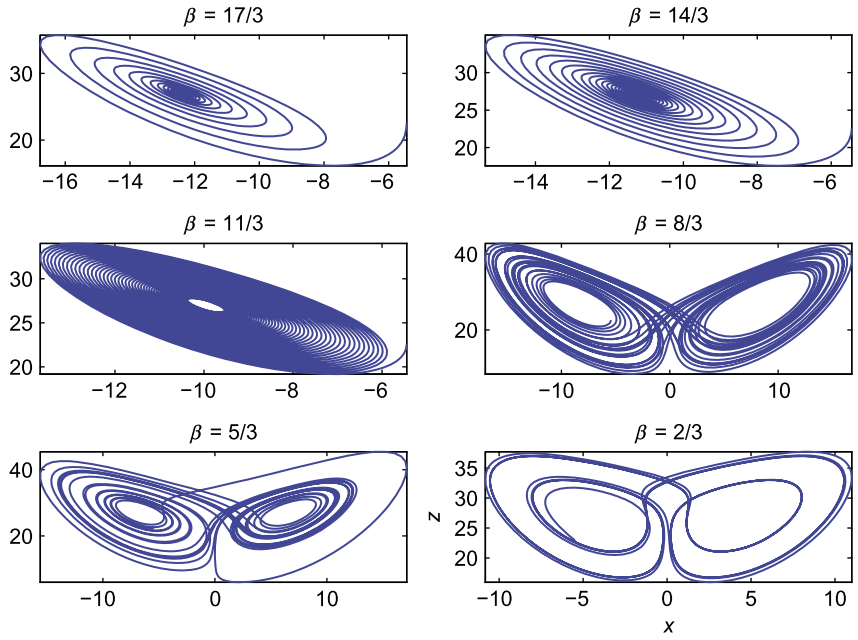


FIGURE 13.5

Plot of the sensitivity of the Lorenz 63 model to the choice of the β parameter.

$$z_{n+1} = z_n + \frac{K_1(z_n) + K_2(z_n)}{2}. \tag{13.39c}$$

The definitions in (13.38a)–(13.38f) are now substituted into (13.39a)–(13.39c). This results in the equations for the discrete nonlinear model as

$$x_{n+1} = \left(1 + \frac{1}{2}\left(h^2\sigma\rho - h^2\sigma z_n - 2h\sigma + h^2\sigma^2\right)\right)x_n + \frac{1}{2}\left(2h\sigma - h^2\sigma - h^2\sigma^2\right)y_n, \tag{13.40a}$$

$$y_{n+1} = \frac{1}{2}\left(2h\rho - h^2\rho\sigma - h^2\rho + \left(h^2\sigma - 2h + h^2 + h^2\beta - h^3\beta\sigma\right)z_n + \left(h^3\sigma - h^2\right)x_n y_n - h^3\sigma y_n^2\right)x_n$$
$$+ \left(1 + \frac{1}{2}\left(h^2\rho\sigma - 2h + h^2 - h^2\sigma + h^3\beta\sigma z_n\right)\right)y_n, \tag{13.40b}$$

$$z_{n+1} = \frac{1}{2}\left(-h^2\sigma + \left(2h - h^2 + h^3\rho\sigma + h^3\sigma - h^2\beta\right)y_n + \left(h^2\rho - h^3\rho\sigma + \left(h^3\sigma - h^2\right)z_n\right)x_n - h^3\sigma y_n z_n\right)x_n$$
$$+ \frac{1}{2}\left(h^2\sigma - h^3\sigma\right)y_n^2 + \left(1 + \frac{1}{2}\left(h^2\beta - 2h\beta\right)\right)z_n. \tag{13.40c}$$

Tangent linear approximations to the Lorenz 63 model

As we stated earlier, there are different ways of finding the tangent linear model. The first is to differentiate (13.37a)–(13.37c) with respect to x, y, and z, and to form the associated tangent linear model matrix. Following this route leads to

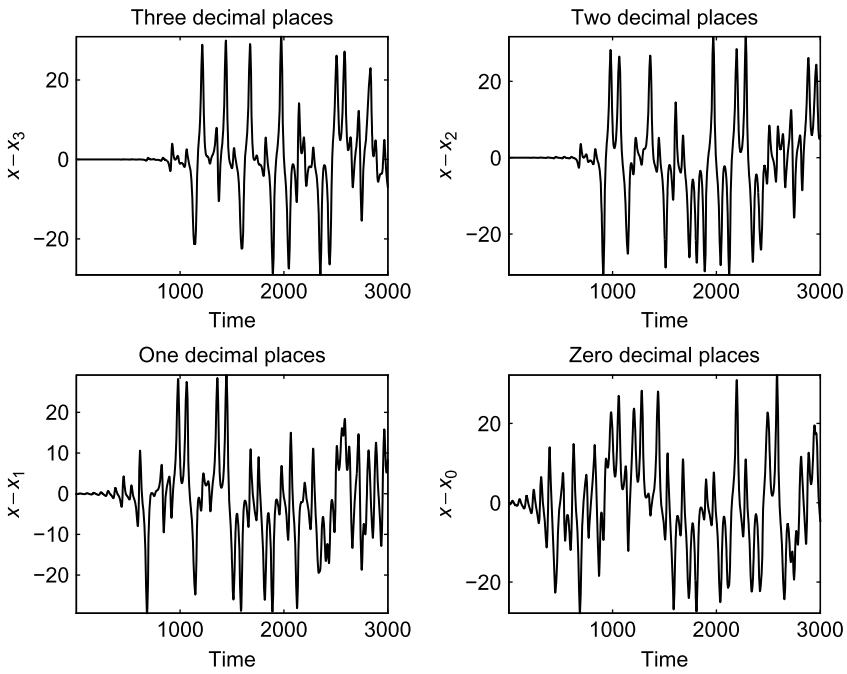


FIGURE 13.6

Plot of the sensitivity of the initial conditions to the Lorenz 63 model.

$$\begin{aligned}
&\frac{\partial \dot{x}}{\partial x} = -\sigma, \quad &&\frac{\partial \dot{x}}{\partial y} = \sigma, \quad &&\frac{\partial \dot{x}}{\partial z} = 0, \\
&\frac{\partial \dot{y}}{\partial x} = \rho - z, \quad &&\frac{\partial \dot{y}}{\partial y} = -1, \quad &&\frac{\partial \dot{y}}{\partial z} = -x, \\
&\frac{\partial \dot{z}}{\partial x} = y, \quad &&\frac{\partial \dot{z}}{\partial y} = x, \quad &&\frac{\partial \dot{z}}{\partial z} = -\beta,
\end{aligned} \tag{13.41}$$

which gives us

$$\begin{pmatrix} \delta\dot{x} \\ \delta\dot{y} \\ \delta\dot{z} \end{pmatrix} = \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho - z & -1 & -x \\ y & x & -\beta \end{pmatrix} \begin{pmatrix} \delta x \\ \delta y \\ \delta z \end{pmatrix}.$$

Expanding the matrix-vector multiplication above yields

$$\begin{aligned}
\delta\dot{x} &= -\sigma\delta x + \sigma\delta y, \\
\delta\dot{y} &= \rho\delta x - \delta y - x\delta z - z\delta x_i, \\
\delta\dot{z} &= y\delta x + x\delta y - \beta\delta z.
\end{aligned} \tag{13.42}$$

The equations above are the analytical version of the tangent linear model. However, it is not always possible to differentiate the nonlinear model equations and so the tangent linear approximation could

be applied instead, which as we saw earlier is $\mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x})$. This is the standard linearization approach, which when applied to the Lorenz 1963 model yields

$$\mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x}) \equiv \begin{cases} \dot{x}+\delta\dot{x} = -\sigma\left[-(x+\delta x)+(y+\delta y)\right], \\ \dot{y}+\delta\dot{y} = \rho(x+\delta x)-y-\delta y-(x+\delta x)(z+\delta z), \\ \dot{z}+\delta\dot{z} = (x+\delta x)(y+\delta y)-\beta(z+\delta z). \end{cases}$$

Now, forming $\mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x})$ results in

$$\mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x}) \begin{cases} \delta\dot{x} = -\sigma\left[-\delta x+\delta y\right], \\ \delta\dot{y} = \rho\delta x-\delta y-x\delta z-z\delta x-\delta x\delta z, \\ \delta\dot{z} = x\delta y+y\delta x+\delta x\delta y-\beta\delta z. \end{cases}$$

Thus we have

$$\begin{aligned} \mathcal{M}(\boldsymbol{x}+\delta\boldsymbol{x})-\mathcal{M}(\boldsymbol{x}) &= \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho-z & -1 & -x \\ y & x & -\beta \end{pmatrix} \begin{pmatrix} \delta x \\ \delta y \\ \delta z \end{pmatrix} + \begin{pmatrix} 0 \\ -\delta x\delta z \\ \delta x\delta y \end{pmatrix}, \\ &= \mathbf{M}\delta\boldsymbol{x}+O\left[(\delta\boldsymbol{x})^2\right], \\ &\equiv \mathbf{L}(t_0,t_i)\,\delta\boldsymbol{x}_0+O\left[(\delta\boldsymbol{x})^2\right]. \end{aligned} \tag{13.43}$$

Therefore, given the tangent linear model equations in (13.42), we again apply the modified Euler scheme to these linear equations to obtain the discrete version of the tangent linear model.

Direct derivation of the adjoint of the modified Euler approximation to the Lorenz 63 model

To form the tangent linear approximation, we must notice that there are 12 variables that we are going to differentiate the nonlinear approximation to the Lorenz 63 model equations with respect to. These 12 variables are

$$\begin{pmatrix} x_n \\ y_n \\ z_n \\ K_1(x_n) \\ K_1(y_n) \\ K_1(z_n) \\ K_2(x_n) \\ K_2(y_n) \\ K_2(z_n) \\ x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{pmatrix}. \tag{13.44}$$

We therefore have 12 active variables but only 3 are unmodified: x_n, y_n, and z_n. The remaining nine variables are functions of the three unmodified variables and the other active variables. We need to add

into the tangent linear model the fact that x_n, y_n, and z_n are not modified, which is equivalent to

$$\begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix}. \tag{13.45}$$

This then yields the rather large matrix-vector equation of

$$\begin{pmatrix} \delta x_n \\ \delta y_n \\ \delta z_n \\ \delta K_{1x} \\ \delta K_{1y} \\ \delta K_{1z} \\ \delta K_{2x} \\ \delta K_{2y} \\ \delta K_{2z} \\ \delta x_{n+1} \\ \delta y_{n+1} \\ \delta z_{n+1} \end{pmatrix} =$$

$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-h\sigma & h\sigma & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
h(\rho - z_n) & -h & -hx_n & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
hy_n & hx_n & -h\beta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-h\sigma & h\sigma & 0 & -h\sigma & h\sigma & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
h\left(\rho - \left(z_n + K_{1z}\right)\right) & -h & -h(x_n + K_{1x}) & h\left(\rho - \left(z_n + K_{1z}\right)\right) & -h & -h(x_n + K_{1x}) & 0 & 0 & 0 & 0 & 0 & 0 \\
h\left(y_n + K_{1y}\right) & h(x_n + K_{1x}) & -h\beta & h\left(y_n + K_{1y}\right) & h(x_n + K_{1x}) & -h\beta & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0
\end{pmatrix}$$

$$\times \begin{pmatrix} \delta x_n \\ \delta y_n \\ \delta z_n \\ \delta K_{1x} \\ \delta K_{1y} \\ \delta K_{1z} \\ \delta K_{2x} \\ \delta K_{2y} \\ \delta K_{2z} \\ \delta x_{n+1} \\ \delta y_{n+1} \\ \delta z_{n+1} \end{pmatrix}.$$

We now need to form the transpose of the matrix above to form the basis of the calculation of the adjoint variables as:

$$
\begin{pmatrix}
\delta x_n^* \\ \delta y_n^* \\ \delta z_n^* \\ \delta K_{1x}^* \\ \delta K_{1y}^* \\ \delta K_{1z}^* \\ \delta K_{2x}^* \\ \delta K_{2y}^* \\ \delta K_{2z}^* \\ \delta x_{n+1}^* \\ \delta y_{n+1}^* \\ \delta z_{n+1}^*
\end{pmatrix} =
$$

$$
\begin{pmatrix}
1 & 0 & 0 & -h\sigma & h(\rho - z_n) & hy_n & -h\sigma & h\left(\rho - \left(z_n + K_{1z}\right)\right) & h\left(y_n + K_{1y}\right) & 1 & 0 & 0 \\
0 & 1 & 0 & h\sigma & -h & hx_n & h\sigma & -h & h\left(x_n + K_{1x}\right) & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & -hx_n & -h\beta & 0 & -h\left(x_n + K_{1x}\right) & -h\beta & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & -h\sigma & h\left(\rho - \left(z_n + K_{1z}\right)\right) & h\left(y_n + K_{1y}\right) & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & h\sigma & -h & h\left(x_n + K_{1x}\right) & 0 & \frac{1}{2} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -h\left(x_n + K_{1x}\right) & -h\beta & 0 & 0 & \frac{1}{2} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
$$

$$
\begin{pmatrix}
\delta x_n^* \\ \delta y_n^* \\ \delta z_n^* \\ \delta K_{1x}^* \\ \delta K_{1y}^* \\ \delta K_{1z}^* \\ \delta K_{2x}^* \\ \delta K_{2y}^* \\ \delta K_{2z}^* \\ \delta x_{n+1}^* \\ \delta y_{n+1}^* \\ \delta z_{n+1}^*
\end{pmatrix}.
$$

This then leads to the nine adjoint equations as

$$\begin{aligned}
\delta x_n^* &= \delta x_n^* - h\sigma\delta K_{1x}^* + h\left(\rho - z_n\right)\delta K_{1y}^* + hy_n\delta K_{1z}^* - h\sigma\delta K_{2x}^* + h\left(\rho - \left(z_n + K_{1z}\right)\right)\delta K_{2y}^* + h\left(y_n + K_{1y}\right) \\
&\quad \times \delta K_{2z}^* + \delta x_{n+1}^*, \\
\delta y_n^* &= \delta y_n^* + h\sigma\delta K_{1x}^* - h\delta K_{1y}^* + h\sigma\delta K_{1z}^* + h\sigma\delta K_{2x}^* - h\delta K_{2y}^* + h\left(x_n + K_{1x}\right)\delta K_{2x}^* + \delta y_{n+1}^*, \\
\delta z_n^* &= \delta z_n^* - hx_n\delta K_{1y}^* - h\beta\delta K_{1z}^* - h\left(x_n + K_{1x}\right)\delta K_{2y}^* - h\beta\delta K_{2z}^* + \delta z_{n+1}^*, \\
\delta K_{1x}^* &= -h\sigma\delta K_{2x}^* + h\left(\rho - \left(z_n + K_{1z}\right)\right)\delta K_{2y}^* + h\left(y_n + K_{1y}\right)\delta K_{2z}^* + \frac{\delta x_{n+1}^*}{2}, \\
\delta K_{1y}^* &= h\sigma\delta K_{2x}^* - h\delta K_{2y}^* + h\left(x_n + K_{1x}\right)\delta K_{2z}^* + \frac{\delta y_{n+1}^*}{2}, \\
\delta K_{1z}^* &= -h\left(x_n + K_{1x}\right)\delta K_{2y}^* - h\beta\delta K_{2z}^* + \frac{\delta z_{n+1}^*}{2}, \\
\delta K_{2x}^* &= \frac{\delta x_{n+1}^*}{2}, \\
\delta K_{2y}^* &= \frac{\delta y_{n+1}^*}{2}, \\
\delta K_{2z}^* &= \frac{\delta z_{n+1}^*}{2}.
\end{aligned}$$

As we can see from the equations above we would have to work backwards to find the values for the adjoint variables δx_n^*, δy_n^* and δz_n^*. Therefore to make the tangent linear model equations flow better we could have rearrange the large vector to have the δK_2s first, then the K_1s, followed by the δx^{n+1}, y^{n+1} and z^{n+1} and then have the equations for the unmodified variables last. If you were to do this then the equations above would be in reverse order, but still the same quantities; however the order of the calculations would flow down the page.

It is quite often the case that it is better to differentiate the code rather than the original differential equations. To highlight this we now will differentiate a version of coding for the modified Euler scheme applied to the Lorenz 1963 model.

Differentiating the code of the modified Euler approximation to the Lorenz 63 model

To find the adjoint of the tangent linear model, we are going to differentiate the code as set out in [316].

In order to code this adjoint correctly, we use the definitions as set out in (13.38a)–(13.39c). The first set of adjoint equations come from (13.39c), where we have the three adjoint variables: $\hat{z}$, $\widehat{K_1(z)}$, and $\widehat{K_2(z)}$. Given these adjoint variables, the adjoint equations for (13.39c) are

$$\hat{z}_i = \hat{z}_i + \delta z_{i+1}, \tag{13.46a}$$

$$\widehat{K_1(z)} = \frac{1}{2}\delta z_{i+1}, \tag{13.46b}$$

$$\widehat{K_2(z)} = \frac{1}{2}\delta z_{i+1}. \tag{13.46c}$$

This same approach is used for (13.39b), where the adjoint variables are $\hat{y}$, $\widehat{K_1(y)}$, and $\widehat{K_2(y)}$ and the adjoint equations are

$$\hat{y}_i = \hat{y}_i + \delta y_{i+1}, \tag{13.47a}$$

$$\widehat{K_1(y)} = \frac{1}{2}\delta y_{i+1}, \tag{13.47b}$$

$$\widehat{K_2(y)} = \frac{1}{2}\delta y_{i+1}, \tag{13.47c}$$

and for (13.39a) with the adjoint variables $\hat{x}$, $\widehat{K_1(x)}$, and $\widehat{K_2(x)}$, the adjoint equations are given by

$$\hat{x}_i = \hat{x}_i + \delta x_{i+1}, \tag{13.48a}$$

$$\widehat{K_1(x)} = \frac{1}{2}\delta x_{i+1}, \tag{13.48b}$$

$$\widehat{K_2(x)} = \frac{1}{2}\delta x_{i+1}. \tag{13.48c}$$

The next set of equations we consider are (13.38f)–(13.38d). For (13.38f) the perturbation of the adjoint variable is $\delta\widehat{K_2(z)}$. Therefore for (13.38f) the derivation requires the derivatives of (13.38f) with respect to x_i, y_i, z_i, $\widehat{K_1(x)}$, $\widehat{K_1(y)}$, and $\widehat{K_1(z)}$. This leads to the adjoint equations as

$$\hat{x}_i = \hat{x}_i + h\left(y_i + k_1(y)\right)\widehat{k_2(z)}, \tag{13.49a}$$

$$\widehat{k_1(x)} = \widehat{k_1(x)} + h\left(y_i + k_1(y)\right)\widehat{k_2(z)}, \tag{13.49b}$$

$$\hat{y}_i = \hat{y}_i + h\left(x_i + \widehat{k_1(x)}\right)\widehat{k_2(z)}, \tag{13.49c}$$

$$\widehat{k_1(y)} = \widehat{k_1(y)} + h\left(z_i + k_1(x)\right)\widehat{k_2(z)}, \tag{13.49d}$$

$$\hat{z}_i = \hat{z}_i - \beta h\widehat{k_2(z)}, \tag{13.49e}$$

$$\widehat{k_1(z)} = \widehat{k_1(z)} - \beta h\widehat{k_2(z)}, \tag{13.49f}$$

$$\widehat{k_2(z)} = 0.$$

For (13.38e) the adjoint variable is $\widehat{k_2(y)}$, which results in the adjoint equations for (13.38e) as

$$\hat{x}_i = \hat{x}_i + (\rho h - h\left(z_i + k_1(z)\right))\widehat{k_2(y)}, \tag{13.50a}$$

$$\widehat{k_1(x)} = \widehat{k_1(x)} + (\rho h - h\left(z_i + k_1(z)\right))\widehat{k_2(y)}, \tag{13.50b}$$

$$\hat{y} = \hat{y}_i - h\widehat{k_2(y)}, \tag{13.50c}$$

$$\widehat{k_1(y)} = \widehat{k_1(y)} - h\widehat{k_2(y)}, \tag{13.50d}$$

$$\hat{z}_i = \hat{z}_i - h\left(x_i + k_1(x)\right)\widehat{k_2(y)}, \tag{13.50e}$$

$$\widehat{k_1(z)} = \widehat{k_1(z)} - h\left(x_i + k_1(x)\right)\widehat{k_2(y)}, \tag{13.50f}$$

$$\widehat{k_2(y)} = 0.$$

Applying the same technique for (13.38d) with the adjoint variable $\widehat{k_2(x)}$ results in

$$\hat{x}_i = \hat{x}_i - h\sigma\widehat{k_2(x)}, \tag{13.51a}$$

$$\widehat{k_1(x)} = \widehat{k_1(x)} - h\sigma\widehat{k_2(x)}, \tag{13.51b}$$

$$\hat{y}_i = \hat{y}_i + h\sigma\widehat{k_2(x)}, \tag{13.51c}$$

$$\widehat{k_1(y)} = \widehat{k_1(y)} + h\sigma\widehat{k_2(x)}, \tag{13.51d}$$
$$\widehat{k_2(x)} = 0.$$

The last set of equations that have to be considered for the model are (13.38a)–(13.38c). Starting with (13.38c) with the adjoint variable as $\widehat{k_1(z)}$ gives us the following adjoint equations:

$$\hat{x}_i = \hat{x}_i + hy_i\widehat{k_1(z)}, \tag{13.52a}$$
$$\hat{y}_i = \hat{y}_i + hx_i\widehat{k_1(z)}, \tag{13.52b}$$
$$\hat{z}_i = \hat{z}_i - h\beta\widehat{k_1(z)}, \tag{13.52c}$$
$$\widehat{k_1(z)} = 0.$$

Again, following the same technique for (13.38b) with the adjoint variable, $\widehat{k_1(y)}$, gives us

$$\hat{x}_i = \hat{x}_i + h(\rho - z_i)\widehat{k_1(y)}, \tag{13.53a}$$
$$\hat{y}_i = \hat{y}_i - h\widehat{k_1(y)}, \tag{13.53b}$$
$$\hat{z}_i = \hat{z}_i - hx_i\widehat{k_1(y)}, \tag{13.53c}$$
$$\widehat{k_1(y)} = 0,$$

adjoint equations.

The final set of adjoint equations for the numerical approximations to (13.37a)–(13.37c) come from considering (13.38a) with the adjoint variable $\widehat{k_1(x)}$, which results in the last two adjoint equations as

$$\hat{x}_i = \hat{x}_i - h\sigma\widehat{k_1(x)}, \tag{13.54a}$$
$$\hat{y}_i = \hat{y}_i + h\sigma\widehat{k_1(x)}, \tag{13.54b}$$
$$\widehat{k_1(x)} = 0.$$

It can be shown that if we combine all of the expressions above in this section then we obtain the same expressions for the adjoint variables as we did with the matrix-vector approach.

13.3.2 Eady Model

In the chapter on semi-Lagrangian methods (Chapter 10), we introduced the quasigeostrophic potential vorticity model referred to as the Eady model. Recalling the system of equations that govern this model, we have

$$q' = \nabla_h^2\psi + \frac{\partial}{\partial z}\left(\frac{f_0^2}{N^2}\frac{\partial\psi}{\partial z}\right),$$
$$D_g b' = -wN^2,$$

where in this model we have the Laplacian operator as well as the Eulerian/Lagrangian form for advection.

If we take the continuous form of the advection equation for the buoyancy

$$\frac{\partial b}{\partial t} + u\frac{\partial b}{\partial x} = \frac{\partial \psi}{\partial x}, \tag{13.55}$$

and introduce perturbations to b, u and to ψ in the forms $b = b_0 + \delta b$ and $\psi = \psi_0 + \delta\psi$. Substituting these expressions above into (13.55) results in

$$\frac{\partial (b_0 + \delta b)}{\partial t} + u\frac{\partial (b_0 + \delta b)}{\partial x} = \frac{\partial (\psi_0 + \delta\psi)}{\partial x}. \tag{13.56}$$

As we presented earlier in Chapter 10, to obtain an approximation to the tangent linear model we use the fact that $f(x + \delta x) - f(x) \approx \frac{\partial f(x)}{\partial x}\delta x$. Therefore the tangent linear approximation to the advection equation in (13.55) is

$$\frac{\partial \delta b}{\partial t} + u\frac{\partial \delta b}{\partial x} = \frac{\partial \delta\psi}{\partial x}. \tag{13.57}$$

The next step is to apply the centered-time, centered-space finite difference scheme to (13.57), which results in

$$\frac{(\delta b)_j^{n+1} - (\delta b)_j^{n-1}}{2\Delta t} = -u_j\frac{(\delta b)_{j+1}^n - (\delta b)_{j-1}^n}{2\Delta x} + \frac{(\delta\psi)_{j+1}^n - (\delta\psi)_{j-1}^n}{\Delta x}. \tag{13.58}$$

We now write (13.58) into matrix form, $\delta\boldsymbol{x} = \mathbf{A}\delta\boldsymbol{x}$ for the six increments in (13.58), which is

$$\begin{pmatrix} \delta\psi_{j+1}^n \\ \delta\psi_{j-1}^n \\ \delta b_{j+1}^n \\ \delta b_{j-1}^n \\ \delta b_j^{n-1} \\ \delta b_j^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ \alpha & -\alpha & -u\alpha & u\alpha & 1 & 0 \end{pmatrix} \begin{pmatrix} \delta\psi_{j+1}^n \\ \delta\psi_{j-1}^n \\ \delta b_{j+1}^n \\ \delta b_{j-1}^n \\ \delta b_j^{n-1} \\ \delta b_j^{n+1} \end{pmatrix}, \tag{13.59}$$

where $\alpha = \frac{\Delta t}{\Delta x}$. To obtain the adjoint expressions for the advection of the buoyancy in the Eady model on the boundaries, we first introduce the adjoint variables $\widehat{\delta\boldsymbol{x}}^T = \left(\widehat{\delta\psi_{j+1}^n} \quad \widehat{\delta\psi_{j-1}^n} \quad \widehat{\delta b_{j+1}^n} \quad \widehat{\delta b_{j-1}^n} \quad \widehat{\delta b_j^{n-1}} \quad \widehat{\delta b_j^{n+1}}\right)$ and taking the transpose of the matrix in (13.59) we obtain the adjoint equations in matrix form as

$$\begin{pmatrix} \widehat{\delta\psi_{j+1}^n} \\ \widehat{\delta\psi_{j-1}^n} \\ \widehat{\delta b_{j+1}^n} \\ \widehat{\delta b_{j-1}^n} \\ \widehat{\delta b_j^{n-1}} \\ \widehat{\delta b_j^{n+1}} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & \alpha \\ 0 & 1 & 0 & 0 & 0 & -\alpha \\ 0 & 0 & 1 & 0 & 0 & -u\alpha \\ 0 & 0 & 0 & 1 & 0 & u\alpha \\ 0 & 0 & 0 & 0 & 1 & a \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \widehat{\delta\psi_{j+1}^n} \\ \widehat{\delta\psi_{j-1}^n} \\ \widehat{\delta b_{j+1}^n} \\ \widehat{\delta b_{j-1}^n} \\ \widehat{\delta b_j^{n-1}} \\ \widehat{\delta b_j^{n+1}} \end{pmatrix}. \tag{13.60}$$

Multiplying out (13.60) results in the adjoint equations for the buoyancy advection on the boundaries as

$$\begin{aligned}
\widehat{\delta\psi^n_{j+1}} &= \widehat{\delta\psi^n_{j+1}} + \alpha\widehat{\delta b^{n+1}_j},\\
\widehat{\delta\psi^n_{j-1}} &= \widehat{\delta\psi^n_{j-1}} - \alpha\widehat{\delta b^{n+1}_j},\\
\widehat{\delta b^n_{j+1}} &= \widehat{\delta b^n_{j+1}} - u\alpha\widehat{\delta b^{n+1}_j},\\
\widehat{\delta b^n_{j-1}} &= \widehat{\delta b^n_{j-1}} + u\alpha\widehat{\delta b^{n+1}_j},\\
\widehat{\delta b^n_{j-1}} &= \widehat{\delta b^n_{j-1}} + \widehat{\delta b^{n+1}_j},\\
\widehat{\delta b^{n+1}_j} &= 0.
\end{aligned}$$

Adjoint of the Laplacian approximation

If we now consider the Laplacian equation for the perturbation of the streamfunction, $\boldsymbol{\psi}$, given the quasigeostrophic potential vorticity, $\boldsymbol{q}$, which is

$$\nabla^2\psi = q, \tag{13.61}$$

subject to the buoyancy boundary conditions, $\frac{\partial\boldsymbol{\psi}}{\partial z} = \boldsymbol{b}$, and the periodicity boundary conditions, then upon applying the five-point stencil we obtain a matrix-vector equation of the form

$$\mathbf{A}\boldsymbol{\psi} = \boldsymbol{d}, \tag{13.62}$$

where

$$\boldsymbol{d} = \begin{pmatrix} \boldsymbol{q}_{bot} + \boldsymbol{b}_{bot} \\ \boldsymbol{q}_{in} \\ \boldsymbol{q}_{top} + \boldsymbol{b}_{top} \end{pmatrix},$$

where *bot* refers to the bottom boundary, *in* refers to the interior of the numerical domain, and *top* refers to the upper boundary.

Therefore, the adjoint of the five-point stencil approximation to (13.61) is the transpose of the matrix-vector equation in (13.62) which is

$$\mathbf{A}^T\widehat{\boldsymbol{d}} = \widehat{\boldsymbol{\psi}}.$$

If we consider the technique to overcome the ill-posedness through adding a small perturbation to the diagonal entries, then the associated $\mathbf{A}$ matrix is symmetric and therefore we have that $\mathbf{A}^T = \mathbf{A}$. The same is not true for the constraining the streamfunction to have a specific point technique, and as such we must be careful in calculating the transpose to take this into account.

13.3.3 Tangent Linear Approximations to Semi-Lagrangian Schemes

When we investigated the accuracy of the numerical approximations to the advection of the quasi-geostrophic potential vorticity through using semi-Lagrangian schemes, we considered different order interpolation schemes to obtain the value of the tracer at the departure point. For the time component of

the advection we considered a simple upwind scheme with a constant velocity, which was appropriate for the formulation of the Eady model; it is quite often not the case.

The paper by Polavarapu et al. [339] provides a rigorous derivation of a consistent approach in deriving the tangent linear model for semi-Lagrangian schemes. In [339] the authors present the initial problem associated with a tangent linear model of a semi-Lagrangian advection scheme by considering Burger's equation with no viscosity, which is given by

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0.$$

They consider a two-time-step semi-Lagrangian scheme of the equation above by

$$u\left(x_i, t^{n+1}\right) = u\left(x_i - \alpha_i, t^n - \Delta t\right), \tag{13.63}$$

where the subscript i refers to the ith grid point and α_i is determined through a second order in time approximation to

$$\frac{dx}{dt} = u\left(x, t\right). \tag{13.64}$$

The velocity, which is the right-hand side of (13.64), needs to be interpolated between grid points if it does not coincide with a grid point. In [339] the authors present an example where there is regular grid spacing combined with linear interpolation so that

$$u\left(x_i - \alpha_i, t^n - \Delta t\right) = au\left(x_{j-1}, t^n - \Delta t\right) + bu\left(x_j, t^n - \Delta t\right), \tag{13.65}$$

where

$$b = \frac{x_i - \alpha_i - x_j - 1}{\Delta x}, \quad a = 1 - b.$$

As we saw in the Eady model example in Chapter 10, we need to determine the integer of the nearest grid point to the apparent departure, or upstream interpolation, point, where, if using FORTRAN, then they state in [339] that the equivalent code line is

$$j = \text{INT}\left(\frac{x_i - \alpha_i}{\Delta x}\right) + 1.$$

However, INT is an intrinsic FORTRAN function, there is an equivalent function in MATLAB®, *floor*, but these steps are not differentiable; while we note that j is a function of the control variable α_i, j is not. Therefore, the tangent linear code is linearized about the determined grid interval, but it should be noted that a perturbed interpolation point **could lie outside this interval**.

The **tangent linear model**, as we have seen, is derived by considering a model that is denoted by $\mathcal{M}(\boldsymbol{x})$ which could be linear, or nonlinear, that updates the state variable $\boldsymbol{x}$ as

$$\boldsymbol{x}^n = \mathcal{M}\left(\boldsymbol{x}^{n-1}\right), \tag{13.66}$$

where n is the time index and $\mathcal{M}$ is a vector valued function of dimensions N, where N is the number of the time steps we require to move the initial conditions $\boldsymbol{x}_0$ forward in time. The next step is to denote

a specific trajectory through time and space by $\overline{\boldsymbol{x}}^n$, for $n = 0, 1, \ldots, N$, that satisfy (13.66); we consider perturbations that are also referred to as variations, about this reference trajectory that evolve according to

$$\delta \boldsymbol{x}^n = \mathcal{M}\left(\overline{\boldsymbol{x}}^{n-1} + \delta \boldsymbol{x}^{n-1}\right) - \mathcal{M}\left(\overline{\boldsymbol{x}}^{n-1}\right). \tag{13.67}$$

We now introduce the *tangent linear hypothesis* which states that for small perturbations, where we mean by small that $|\delta \boldsymbol{x}| << |\overline{\boldsymbol{x}}|$, then a good approximation to (13.67) is given by the linear part of the nonlinear variation

$$\frac{d\mathcal{M}}{d\boldsymbol{x}}\left(\overline{\boldsymbol{x}}^{n-1}\right)\delta \boldsymbol{x}^{n-1}. \tag{13.68}$$

Therefore, by definition the tangent linear model describes the evolution of the linear, sometime referred to as the first order, variation that is tangent to the reference trajectory in phase space. Thus the tangent linear model should contain all of the linear parts of the nonlinear variation and is therefore equal to the expression in (13.68). It is stated in [339] that the hypotheses is true for continuous or theoretical models. However, in numerical geophysical modeling, we have discrete numerical models where the tangent linear model is obtained by differentiating the codes line by line, which is what is done in [253,316], and due to the existence of non-differentiability of discrete programming structures, the tangent linear model is not necessarily equal to the linear variation. This is an important point that needs to be kept in mind when deriving the tangent linear model.

In [339] the authors introduce the concept of **correctness** which they state refers to the ability of the model to asymptote to the linear part of the nonlinear variation as the size of the perturbation tends to zero. Given the property of correctness [339] devises a measure to distinguish between the linear variation predicted by the tangent linear model, $(\delta \boldsymbol{x}^n)^L$ defined as

$$\left(\delta \boldsymbol{x}^n\right)^L = \boldsymbol{L}\left(\boldsymbol{x}^{n-1}\right)\delta \boldsymbol{x}^{n-1}, \tag{13.69}$$

where $\boldsymbol{L}$ is the linear operator, and the linear part of the nonlinear variation which is defined as (13.68). The tangent linear model is correct when it is equal to the linear variation; that is to say, when $\boldsymbol{L} = \frac{d\mathcal{M}}{dx}$. In [339] the authors next define the **linearization error**, which is given by

$$E \equiv \delta \boldsymbol{x}^n - \left(\delta \boldsymbol{x}^n\right)^L = \mathcal{M}\left(\overline{\boldsymbol{x}}^{n-1} + \delta \boldsymbol{x}^{n-1}\right) - \mathcal{M}\left(\overline{\boldsymbol{x}}^{n-1}\right) - \left(\delta \boldsymbol{x}^n\right)^L. \tag{13.70}$$

Expanding the nonlinear model in (13.70) as a Taylor series, and considering the ith component of $\boldsymbol{x}$, it can then be shown that this expansion is equal to

$$E_i = \sum_{j=1}^{N}\left(\frac{\partial \mathcal{M}_i}{\partial x_j}\left(\overline{\boldsymbol{x}}^{n-1}\right) - l_{ij}\left(\overline{\boldsymbol{x}}^{n-1}\right)\right)\delta x_j^{n-1} + \frac{1}{2}\sum_{j=1}^{N}\sum_{k=1}^{N}\frac{\partial^2 \mathcal{M}_i}{\partial x_j \partial x_k}\left(\overline{\boldsymbol{x}}^{n-1}\right)\delta x_j^{n-1}\delta x_k^{n-1} + O\left(\left(\delta \boldsymbol{x}^{n-1}\right)^3\right). \tag{13.71}$$

The l_{ij} term in (13.71) refers to the (i, j)th element of $\boldsymbol{L}$. Thus the tangent linear model is correct if and only if for all i and all $\delta \boldsymbol{x}^{n-1}$, then E_i is second order in terms of the perturbed variables as $\delta \boldsymbol{x}^{n-1}$ tends to zero. The reader is recommended to read [339] or [131] for more details about the interpretation of the linearization error. What is important for us here is how this measure is used to ascertain the tangent linear approximation, and therefore, by association, the adjoint of the semi-Lagrangian schemes.

Linearizing semi-Lagrangian interpolation

In this section we continue to follow the theory set out in [339]. We start by considering a continuously differentiable function $y(x)$ with independent variable x. The interpolation of y at a given point ξ could be described as

$$\eta = (\xi). \tag{13.72}$$

As a result of $y(x)$ being known everywhere, the interpolation has simply become a function evaluation here. However, it is more likely to be the case that y is only known at a finite number of points so that $y(x)$ would be approximated through some interpolating function. The next step is to assume that $y(x)$ and ξ consist of two parts, $y(x) = \overline{y(x)} + \delta y(x)$ and $\xi = \overline{\xi} + \delta\xi$, where the two perturbations just mentioned are small. The tangent linear variation of η due to the variation in the function itself $\delta y(x)$ and the point of interpolation $\delta\xi$ is

$$\delta v^L = \delta y\left(\overline{\xi}\right) + \left.\frac{d\overline{y}}{dx}\right|_{x=\overline{\xi}} \delta\xi. \tag{13.73}$$

In [339], the expression in (13.73) is labeled the *tangent linear equation* as it is not a model. From (13.72) we can see that η is a linear function of $y(x)$, but in general is a nonlinear function of ξ. If we had the case where there was no variation in ξ then the interpolation would be a linear process. Therefore in [339] the authors consider the case where there are only variations in ξ and set $\delta y(x) = 0$ to isolate the nonlinearity of the interpolation and to identify conditions for the linearization. Given these assumptions, it can be shown that the linearization error for this situation is given by

$$E = \overline{y}\left(\overline{\xi} + \delta\xi\right) - \overline{y}\left(\overline{\xi}\right) - \frac{d\overline{y}}{dx}\left(\overline{\xi}\right)\delta\xi. \tag{13.74}$$

Given the expression in (13.74), we now turn our attention to deriving the tangent linear equation for the interpolation operator in the semi-Lagrangian methods, where we have a grid with $N+1$ grid points, that are denoted by $x_0, x_1, \ldots, x_N$, and at these grid points we have values of the function y denoted by $y_0, y_1, \ldots, y_N$, and finally we denote a piecewise interpolation function $P(x)$ that takes the values of y_j at $P\left(x_j\right)$. We now assume that the original interpolation point, $\overline{\xi}$, lies in the half-open interval $\left[x_{j-1}, x_j\right)$ and that the perturbed interpolation point lies in the half-open interval

$$\overline{xi} + \delta\xi \in \left[x_{i-1}, x_i\right). \tag{13.75}$$

We have presented a copy of the figure from [339] to provide an illustration of where the perturbed departure point could end up, relative to the departure point in Fig. 13.7.

We now consider the nonlinear variation that is determined by

$$P_i\left(\overline{\xi} + \delta\xi\right) - P_j\left(\overline{\xi}\right), \tag{13.76}$$

where the subscripts refer to the interval for which the interpolation functions is appropriate. Therefore, the tangent linear equation for this situation is

$$\delta P_j\left(\overline{\xi}\right) = \frac{dP_j}{d\xi}\left(\overline{\xi}\right)\delta\xi. \tag{13.77}$$

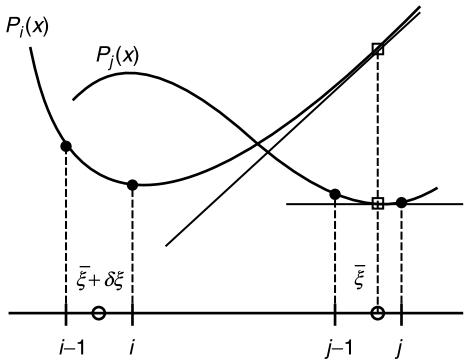


FIGURE 13.7

Copy of figure 1 from Saroja Polavarapu, Monique Tanguay, Richard Mànard & Andrew Staniforth (1996) The tangent linear model for semi-Lagrangian schemes: linearizing the process of interpolation, Tellus A: Dynamic Meteorology and Oceanography, 48:1, 74-95, DOI: 10.3402/tellusa.v48i1.11633. https://creativecommons.org/licenses/by/4.0/.

Given the expression of the tangent linear model for the semi-Lagrangian interpolation operator, we now consider the case from [339] where a general interpolation scheme is given by

$$P(x) = a_j y_{j-1} + b_j y_j + \frac{h_j^2}{6}\left(\left(a_j^3 - a_j\right) y''_{j-1} + \left(b_j^3 - b_j\right) y''_j\right), \tag{13.78}$$

where

$$a_j = \frac{x_j - x}{h_j}, \quad b_j = \frac{x - x_{j-1}}{h_j}, \quad h_j = x_j - x_{j-1}, \tag{13.79}$$

and the expressions for how the second derivatives of y are approximated determines the order and type of interpolation.

The next step in the derivation of the tangent linear model of the semi-Lagrangian scheme is to differentiate the general interpolation formula in (13.79) with respect to $\bar{\xi}$. The differentiation of the polynomial is achieved through the chain rule and by evaluating (13.78) and (13.79) with $x = \bar{\xi}$. This results in the tangent linear model being

$$\delta P_j(\bar{\xi}) = \left(\frac{\bar{y}_j - \bar{y}_{j-1}}{h_j} - \frac{h_j}{6}\left(3\bar{a}_j^2 - 1\right)\bar{y}''_{j-1} - \left(3\bar{b}_j^2 - 1\right) y''_j\right)\delta\xi, \tag{13.80}$$

where

$$a_j = \frac{x_j - \bar{\xi}}{h_j}, \quad b_j = \frac{\bar{\xi} - x_{j-1}}{h_j}. \tag{13.81}$$

We now move on to consider the work in Tanguay and Polavarapu [427], which is where the tangent linear and the adjoint for the cubic Lagrange interpolation are derived.

Tanguay and Polavarapu [427]

In [427] the authors apply the theory from [339] to a 1D passive tracer equation on the sphere in the longitudinal direction, which is defined as

$$\frac{\partial F}{\partial t} + \omega(\lambda)\frac{\partial F}{\partial \lambda} = 0, \tag{13.82}$$

where $F = F(\lambda, t)$ represents the tracer field at time t over the interval $\lambda \in [0, 2\pi]$, $\omega(\lambda)$ is the angular velocity which is greater than zero. Since the equation in (13.82) is linear with respect to the tracer, and given that the authors do not allow any variation in the wind, the tangent linear model and the nonlinear model are the same equation.

As we saw in Chapter 10, the Lagrangian property of (13.82) states that if (13.82) is integrated forward one time step, then the field at the arrival point λ at time $t + \Delta t$ is equal to the value of the tracer at the departure point $\xi(\lambda)$ at time t, through

$$F(\lambda, t + \delta t) = F(\xi(\lambda), t), \tag{13.83}$$

where

$$\int_{\xi(\lambda)}^{\lambda} \frac{d\overline{\lambda}}{\omega(\overline{\lambda})} = \Delta t. \tag{13.84}$$

Note: it is highly likely that the departure, or upstream, point coincides with a grid point and as such there will be a need for an interpolation of the function to that point.

In the theory presented for the adjoints, we know that we require an inner product. In [427] the authors define their inner product to be

$$\langle F, G\rangle = \int_0^{2\pi} F(\lambda) G(\lambda) d\lambda. \tag{13.85}$$

As the formulation of the adjoint in [427] is associated with 4D VAR, which we have not defined yet, they refer to a forcing term $\hat{F}_T(\lambda)$, at the end of the assimilation period of length T, which makes the adjoint equation associated with (13.82)

$$\frac{\partial \hat{F}}{\partial t} + \frac{\partial}{\partial \lambda}\left(\omega(\lambda)\hat{F}\right) = \frac{\partial \hat{F}}{\partial t} + \frac{d\omega(\lambda)}{dx}\hat{F} + \omega(\lambda)\frac{\partial \hat{F}}{\partial x} = 0, \tag{13.86}$$

where $\hat{F} \equiv \hat{F}(\lambda, t)$ represents the adjoint variable with $\hat{F}_T(\lambda, T) = \hat{F}_T(\lambda)$. This give rise to the conservative form of the passive tracer equation. Now if (13.86) is integrated backward for one time step, then the adjoint field $\hat{F}$ at the departure point λ at time t is given in terms of the adjoint variable $\hat{F}$ at the arrival point $\hat{\xi}(\lambda)$ at time $t + \Delta t$ as

$$\hat{F}(\lambda, t) = \hat{F}\left(\hat{\xi}(\lambda), t + \Delta t\right)\frac{\omega\left(\hat{\xi}(\lambda)\right)}{\omega(\lambda)}, \tag{13.87}$$

where

$$\int_{\lambda}^{2\pi} \frac{d\overline{\lambda}}{\omega(\overline{\lambda})} = \Delta t. \tag{13.88}$$

Generally the upstream point position $\xi = \xi(\lambda)$ defined in (13.84) will lie between grid points and so the evaluation of $F(\xi(\lambda), t)$ in (13.83) will require an interpolation to be applied as we only have values of F at the grid points when we are considering the numerical approximation to (13.82). The piecewise continuous interpolation to obtain the value of F in the semi-Lagrangian formulation at the point $\lambda_i = i\Delta\lambda$ is given by

$$F_i(t + \Delta t) = P_{i-p}(\xi), \tag{13.89}$$

where P_{i-p} is the interpolating functions over the interval $(i - p, i - p - 1)$ and $F_i = F(\lambda_i)$. If we recall the general definition for the interpolation scheme, then we obtain

$$P_{i-p}(\xi) = aF_{i-p-1} + (1 - a)F_{i-p} + \frac{\Delta\lambda^2}{6}\left(xF''_{i-p-1} + yF''_{i-p}\right),$$

where for the cubic Lagrange interpolation we have

$$F''_i = \frac{F_{i-1} - 2F_i + F_{i+1}}{\Delta\lambda^2}\Delta\lambda^2. \tag{13.90}$$

As we saw in Chapter 10, the index $i - p$ refers to the interval $[(i - p - 1)\Delta x, (i - p)]$. We recall that the Courant number is calculated through $\alpha = \alpha(\lambda) = \dfrac{\lambda - \xi(\lambda)}{\Delta\lambda}$, and p is the largest integer value such that $p = \int(K)$ where $p \leq K$. The displacement between the two grid points is denoted as $a = a(\lambda)$ such that

$$\xi = (i - p)\Delta\lambda - a\Delta\lambda, \tag{13.91}$$

where $x = \left(a^3 - a\right)$ and $y = \left((1 - a)^3 - (1 - a)\right)$.

It is possible to write all of these equations for the cubic Lagrange interpolation in matrix vector form as

$$\mathbf{F}(t + \delta t) = \mathbf{M}\mathbf{F}(t), \tag{13.92}$$

where the matrix $\mathbf{M}$ is defined as

$$\mathbf{M} = \mathbf{I}_a\mathbf{S}_{p+1} + \mathbf{I}_{(1-a)}\mathbf{S}_p + \frac{\Delta\lambda^2}{6}\left(\mathbf{I}_x\mathbf{S}_{p+1} + \mathbf{I}_y\mathbf{S}_p\right)\mathbf{D}_2, \tag{13.93}$$

and the $\mathbf{I}_z$s, for $z = a, (1 - a)$, x and y, are diagonal matrices with the definitions for the values of z on the diagonal entries, $\mathbf{S}$ is the p-shift operator such that the entries at $(i, i - p(\lambda_i))$ is equal to 1, while all the other entries are equal to 0, and finally $\mathbf{D}_2$ is the numerical approximation to the second derivative operator.

The adjoint of (13.92), with respect to a discretized form of (13.84), is given by

$$\langle F, G\rangle = \Delta\lambda\sum_{i=1}^{N} F_i G_i, \tag{13.94}$$

is

$$\widehat{\mathbf{F}}(t) = \mathbf{M}^T\widehat{\mathbf{F}}(t + \Delta t), \tag{13.95}$$

where

$$\mathbf{M}^T = \mathbf{S}_{p+1}^T \mathbf{I}_a + \mathbf{S}_p^T \mathbf{I}_{(1-a)} + \frac{\Delta\lambda^2}{6}\mathbf{D}_2 \left(\mathbf{S}_{p+1}^T \mathbf{I}_x + \mathbf{S}_p^T \mathbf{I}_y\right). \tag{13.96}$$

The transposed shift operator matrix, $\mathbf{S}_p^T$, depends upon the variation in the integer value p of the Courant number as a function of the grid point. The simplest case is when p is the same for all grid points. Under these circumstances then $\mathbf{S}_p^T = \mathbf{S}_{-p}$, which shares the eigenvalue basis of $\mathbf{D}_2$ [427]. If this is the case, then from (13.94) and (13.95) we have that the numerical adjoint at the grid point λ_i is given by

$$\hat{F}_i(t) = \left((1-a)\hat{F}\right)_{i+p} + \left(a\hat{F}\right)_{i+p+1} + \frac{\Delta\lambda^2}{6}\left(\left(y\hat{F}\right)''_{i+p} + \left(x\hat{F}\right)''\right), \tag{13.97}$$

where $\hat{F}$ on the right-hand side of the equation in (13.97) is evaluated at $t + \Delta t$. An important feature to note here is that if the angular velocity is constant then $\left(a\hat{F}\right)_{i+p+1} = a\left(\hat{F}\right)_{i+p+1}$. Therefore, from (13.97) we can say that the numerical adjoint of the 1D passive tracer equation is a backward integration of a passive tracer problem starting from the forcing $\hat{F}_T(\lambda)$ at time T and moving downstream to the flow using a semi-Lagrangian formulation with the same interpolation scheme as the forward integration.

There are many different versions of the adjoint of the semi-Lagrangian method that can occur due to variations to many different parts of the equation. The reader is referred to [427] for a more detailed explanation of implementing the adjoint method for this problem for these different formulations, as well as [131].

13.3.4 Adjoint of Spectral Transforms

We recall from Chapter 12 that model variables can be represented in spectral space and grid point space, where in spectral space the equations for these variables are expressed in terms of spherical harmonic transforms. We showed that spherical harmonics are the associated Legendre polynomials $P_n^m(\theta)$ multiplied by the complex Fourier series $e^{im\lambda}$. Recall that the subscript n is the total wavenumber and the superscript m is the zonal wavenumber. We assume that the truncation of the Fourier series is triangular, with M indicating the total number of resolvable waves. Now if we denote the set of complex spherical harmonic coefficients for a variable x by $S_n^m(t)$, then the spectral to grid expansion of x is

$$x\left(\lambda_i, \theta_j, \eta_k, t\right) = \sum_{m=-M}^{M}\left(\sum_{|m|}^{M} S_n^m\left(\eta_k, t\right) P_n^m\left(\theta_j\right)\right) e^{im\lambda_i}, \tag{13.98}$$

where i and j represent the discrete grid longitude and latitude, respectively, of the transform grid, while η_k are discrete vertical coordinate values.

The grid point to spectral transform, as we showed in Chapter 12, is obtained from (13.98) by applying the orthogonal property of spherical harmonics:

$$S_n^m(\eta, t) = \frac{1}{2\pi}\int_{-1}^{1} P_n^m(\theta)\left(\int_0^{2\pi} x(\theta, \eta, t)\, e^{-im\lambda} d\lambda\right) d\theta. \tag{13.99}$$

The expression inside the brackets is a forward Fourier transform, which we know can be solved through a fast Fourier transform approach. After applying the fast Fourier transform (13.99) becomes

$$S_n^m(\eta,t)=\int_{-1}^{1}\mathcal{F}^m(x(\theta,\eta,t))\,P_n^m(\theta)\,d\theta, \tag{13.100}$$

where $\mathcal{F}^m(x(\theta,\eta,t))$ are the coefficients from the Fourier transform.

The integration in (13.100) is performed using the Gaussian quadrature method, where, given a polynomial, $f(\theta)$, of degree, $2K-1$ or less, the integral of f from -1 to 1 is computed exactly as

$$\int_{-1}^{1} f(\theta)\,d\theta=\sum_{l=1}^{L} w_l f(\theta_l),$$

where θ_l are the Gaussian latitudes and w_l are the Gaussian quadrature weights. Given these discrete approximations, we can define the discrete form of (13.100) as

$$S_n^m(\eta_k,t)=\sum_{j=1}^{\frac{3M+1}{2}} w_j\mathcal{F}\left(x\left(\theta_j,\eta_k,t\right)\right)P_n^m\left(\theta_j\right), \tag{13.101}$$

where the summation is over $\frac{3M+1}{2}$, which is the number of Gaussian latitudes required to ensure that the integral in (13.100) is evaluated exactly.

The first step in deriving the adjoint of (13.99) is to notice that (13.99) can be written as a matrix-vector equation of the form

$$\begin{pmatrix} S_1 \\ S_2 \\ \vdots \\ S_{M^2} \end{pmatrix}=\mathbf{K}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{mn} \end{pmatrix}, \tag{13.102}$$

where $\mathbf{K}$ is the linear operator that is comprised of the discrete Fourier transforms and the Gaussian quadratures which converts a horizontal grid point field of mn degrees of freedom to a set of complex spherical harmonic coefficients with M^2 degrees of freedom. Therefore, from our definition of the adjoint we have that

$$\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \vdots \\ \hat{x}_{mn} \end{pmatrix}=\mathbf{K}^T\begin{pmatrix} \hat{S}_1 \\ \hat{S}_2 \\ \vdots \\ \hat{S}_{M^2} \end{pmatrix}. \tag{13.103}$$

Therefore, it can be shown that the adjoint of the grid point space to spectral space transform is

$$\hat{x}\left(\lambda_i,\theta_j\right)=w_j\sum_{m=-M}^{M}\left(\sum_{n=|m|}^{M}\hat{S}_n^m P_n^m\left(\theta_j\right)\right)e^{im\lambda}, \tag{13.104}$$

which is the spectral space to grid point space transform of the forward model with an additional scaling factor weighting each Gaussian latitude of the horizontal grid after the transform.

It is also possible through using similar agreement to show that the adjoint of the spectral space to grid point space transform is given by

$$\hat{S}_n^m = \sum_{j=1}^{\frac{3M+1}{2}} \mathcal{F}\left(\hat{x}\left(\theta_j\right)\right) P_n^m\left(\theta_j\right), \tag{13.105}$$

which is similar to (13.101) but without the Gaussian quadrature weights. For a more detailed derivation of differential operators and their adjoints in spectral space the reader is referred to Rosmond [370].

13.4 Perturbation Forecast Modeling

So far in this chapter, we have presented the theory and examples for tangent linear modeling and adjoints have been through the linearization of the discrete version of the equations. In [243] the authors refer to obtaining the linear model through linearizing the discrete linear model as the *tangent linear model*, where the adjoint is found through the transposition of the matrix that represents the discrete tangent linear model. As we have seen this approach had the advantage that the tangent linear model can be found by directly linearizing the nonlinear model source code, which is achieved by transposing the tangent linear source code [243]. This process is known as *automatic differentiation*, which can be performed by hand or through using automatic differentiation compiler tools that are readily available.

An alternative approach that was developed at the United Kingdom's Meteorological Office to be used with their operational 4D incremental VAR data assimilation system involved taking the continuous equations of the nonlinear model and to first linearize these to form a set of linear equations. It was these linearized equations that were then discretized with some suitable numerical scheme, to form the discrete linear model which is referred to as the **perturbation forecast model**. Given this approach, then the adjoint model was obtained from the perturbation forecast model by a transposition of the perturbation forecast models' source code. It is shown in [240,262] that a 4D incremental VAR scheme that uses this approach still contains the exact adjoint of the discrete linear model. In [243] the authors refer to this process of deriving the adjoint this way as a *semi-continuous* method. The reason for the designation of semi-continuous was to differentiate this approach from the *continuous approach* of obtaining an adjoint model; this is where the adjoint model is coded directly from the continuous adjoint equations.

In [243] the authors state two advantages of the perturbation forecast model over the discrete tangent linear model approach. We summarize these advantages here:

(1) The first advantage is based upon the premise that, although the tangent linear model is valid for infinitesimal perturbations, it is the finite perturbations that are comparable to the size of uncertainties in the initial conditions [106]. A perturbation forecast model can be designed to be based upon physical principles and be able to make some small approximations to the true tangent linear model. The semi-continuous approach to developing linear model allows such approximations to be made to both the equations of the linear model, before any discretization, which could be achieved through scale analysis, and in the implementation of the numerical scheme. As a result of this approach savings can be made in the execution costs of the linear model and its adjoint.

As numerical models become more complex, such approximations are likely to become more important, since a direct linearization of the discrete nonlinear model is likely to lead to many small terms that are costly to evaluate but that also do not add much information to the data assimilation process.

(2) A second advantage of the semi-continuous approach is that it is possible to avoid some of the problems that occur when linearizing complex systems. We have stated in the summary of the work in [339] [427] that problems occur in the direct linearization of the interpolation with the semi-Lagrangian advection schemes. By forming the linear model form of the continuous linear equations, many of these difficulties are avoided.

13.4.1 Example With a 1D Shallow Water Equations Model

In [243] the authors present an example of the differences in the derivation, and the performance, of the perturbation forecast model and the tangent linear approximation, where this example is the one-dimensional shallow water equations model. A more detailed study of the two techniques can be found in Lawless [241].

The continuous nonlinear equations used in [243] were those that describe a one-dimensional shallow water system where the flow is of a single layer fluid over an obstacle in the absence of rotation. The associated system of nonlinear partial differential equations is given by

$$\frac{Du}{dt} + \frac{\partial \phi}{\partial x} = -g\frac{\partial H}{\partial x}, \tag{13.106a}$$

$$\frac{D(\ln \phi)}{Dt} + \frac{\partial u}{\partial x} = 0, \tag{13.106b}$$

where

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + u\frac{\partial}{\partial x}, \tag{13.107}$$

is the material derivative, $H = H(x)$, is the height of the bottom orography, u is the velocity of the fluid, and $\phi = gh$ is the geopotential, where g is the gravitational constant, and h, which is greater than zero, is the depth of the fluid above the orography. The problem is defined on the domain $x \in [0, L]$ and $t \in [0, T]$. The spatial boundary conditions are taken to be periodic for the three fields,

$$u(0,t) = u(L,t), \quad \phi(0,t) = \phi(L,t), \quad H(0) = H(L).$$

The initial conditions for the wind and the geopotential fields across the whole domain are given by

$$u(x,0) = u_0(x), \quad \phi(x,0) = \phi(x).$$

Linearized equations

The linearized versions of (13.106a) and (13.106b) are formed through considering the fields u and ϕ and perturbations δu and $\delta\phi$ about a spatially and temporally varying basis state $\overline{u}$ and $\overline{\phi}$, that satisfy the nonlinear equations. Thus we have

$$u(x,t) = \overline{u}(x,t) + \delta u(x,t), \tag{13.108a}$$

$$\phi(x,t) = \overline{\phi}(x,t) + \delta\phi(x,t). \tag{13.108b}$$

Substituting (13.108a) and (13.108b) into (13.106a), the momentum equation, and (13.106b), the continuity equation, yields

$$\frac{D\delta u}{Dt} + \delta u \frac{\partial \overline{u}}{dx} + \frac{\partial \delta\phi}{\partial x} = 0, \tag{13.109a}$$

$$\frac{D}{DT}\left(\frac{\delta\phi}{\overline{\phi}}\right) + \delta u \frac{\partial \ln \overline{\phi}}{\partial x} + \frac{(\delta u)}{\partial x} = 0, \tag{13.109b}$$

where the material derivative $\frac{D}{Dt}$ is defined as in (13.108a), whereas the velocity is the linearization state $\overline{u}$.

Numerical models

The numerical approximation that is used in [243] is a two-time-level semi-implicit semi-Lagrangian scheme [434], but with an off-centered time averaging of the forcing term along the trajectory. An Arakawa C-grid was used for the numerical modeling.

Nonlinear model: The arrival and departure points for the wind and geopotential height fields are denoted by au, $a\phi$, du, and $d\phi$, respectively. The time discretization of the nonlinear model is then

$$\frac{u_{au}^{n+1} - u_{du}}{\Delta t} + (1-\alpha_1)\left(\frac{\partial\phi}{\partial x} + g\frac{\partial H}{\partial x}\right)_{du}^{n} + \alpha_1\left(\frac{\partial\phi}{\partial x} + g\frac{\partial H}{\partial x}\right)_{au}^{n+1} = 0, \tag{13.110a}$$

$$\frac{(\ln\phi)_{a\phi}^{n+1} - (\ln\phi)_{d\phi}^{n}}{\Delta t} + (1-\alpha_2)\left.\frac{\partial u}{\partial x}\right|_{d\phi}^{n} + \alpha_2\left.\frac{\partial u}{\partial x}\right|_{a\phi}^{n+1} = 0, \tag{13.110b}$$

where the superscript is the discrete time index, and the coefficients α_1 and α_2 are time weighting parameters chosen to line in the interval [0.5, 1]. When $\alpha_1 = \alpha_2 = 0.5$, the scheme is second order in time. The procedure for implementing this numerical scheme can be found in [243].

Tangent linear model: The tangent linear model that is used in [243] is obtained by differentiating the nonlinear model's source code. The non-differentiable procedures within the semi-Lagrangian scheme are treated by assuming that the perturbations do not move the departure point outside of the grid interval defined by the linearization state. The interpolation scheme used in all three models is the cubic Lagrange interpolation scheme.

Perturbation forecast model: The perturbation forecast model is developed through the continuous linear equations (13.109a) and (13.109b). A comparison of these equations with the nonlinear model equations in (13.106a) and (13.106b) indicates that they have the same structure, except that the linear equations have an extra term in which the wind field perturbation multiplies the gradient of the linearization state. These terms are discretized as a semi-implicit semi-Lagrangian with the off-centered averages along the trajectory. The remaining terms in (13.109a) and (13.109b) are discretized as the corresponding terms in the nonlinear model. Therefore, the numerical equations for the perturbation forecast model for this example are

$$\frac{\delta u_{au}^{n+1} - \delta u_{du}^{n}}{\Delta t} + (1-\alpha_1)\left.\frac{\partial\delta\phi}{\partial x}\right|_{du}^{n} + \alpha_1\left.\frac{\partial\delta\phi}{\partial x}\right|_{au}^{n+1} + (1-\alpha_3)\left(\delta u\frac{\partial\overline{u}}{\partial x}\right)_{du}^{n} + \alpha_3\left(\delta u\frac{\partial\overline{u}}{\partial x}\right)_{au}^{n+1} = 0, \tag{13.111a}$$

$$\frac{1}{\Delta t}\left(\left(\frac{\delta\phi}{\overline{\phi}}\right)_{au}^{n+1}-\left(\frac{\delta\phi}{\overline{\phi}}\right)_{d\phi}^{n}\right)+(1-\alpha_2)\left.\frac{\partial\delta u}{\partial x}\right|_{d\phi}^{n}+\alpha_2\left.\frac{\partial\delta u}{\partial x}\right|_{a\phi}^{n+1}$$
$$+(1-\alpha_4)\left(\delta u\frac{\partial\left(\ln\overline{\phi}\right)}{\partial x}\right)_{d\phi}^{n}+\alpha_4\left(\delta u\frac{\partial\left(\ln\overline{\phi}\right)}{\partial x}\right)_{a\phi}^{n+1}=0, \tag{13.111b}$$

where α_i, for $i=1,2,3,4$ are time-weighting coefficients. As with the discrete nonlinear model and the tangent linear model, if the time weights are all set to 0.5 then the numerical approximations is second order in time.

We now move on to consider some properties of adjoints, mainly in the form of adjoint sensitivities, singular vectors, and observation impact.

13.5 Adjoint Sensitivities

The starting point for adjoint sensitivity is to consider a function of the output parameters of our model. We shall denote the output of the model by $\boldsymbol{y}$, and the function of the outputs as $J(\boldsymbol{y})$. It is assumed that the function J is not too complicated, so that an analytical expression for the gradient with respect to $\boldsymbol{y}$, denoted $\nabla_{\boldsymbol{y}}J$ is obtainable. Next we introduce an inner product with respect to $\boldsymbol{y}$, denoted by $\langle\cdot,\cdot\rangle_{\boldsymbol{y}}$, which is defined in the space of the output parameters.

From the definition of $\nabla_{\boldsymbol{y}}J$, then for any perturbation to the output parameters, $\delta\boldsymbol{y}$, to first order is given by

$$\delta J=J(\boldsymbol{y}+\delta\boldsymbol{y})-J(\boldsymbol{y})=\left\langle\nabla_{\boldsymbol{y}}J,\delta\boldsymbol{y}\right\rangle. \tag{13.112}$$

Recalling from the introduction to adjoints earlier that we can relate a change in the outputs to a change in the inputs through

$$\delta\boldsymbol{y}=\mathbf{M}\delta\boldsymbol{x},$$

enables (13.112) to be written as

$$\delta J=\left\langle\nabla_{\boldsymbol{y}}J,\mathbf{M}_{\boldsymbol{x}}\delta\boldsymbol{x}\right\rangle_{\boldsymbol{y}}. \tag{13.113}$$

If we now introduce an inner product with respect to $\boldsymbol{x}$, which is over the space of the inputs to the numerical model, then we have

$$\delta J=\left\langle\mathbf{M}^T\nabla_{\boldsymbol{y}}J,\delta\boldsymbol{x}\right\rangle_{\boldsymbol{x}}, \tag{13.114}$$

which implies that we can relate the gradient of the function J with respect to the outputs $\boldsymbol{y}$ to the gradient of the function with respect to the inputs $\boldsymbol{x}$ as

$$\nabla_{\boldsymbol{x}}J=\mathbf{M}^T\nabla_{\boldsymbol{y}}J. \tag{13.115}$$

The expression in (13.115) is a more mathematical definition for the sensitivity derived in the introduction section of this chapter.

Adjoint sensitivity studies are quite often used to determine sensitivities to the initial conditions of a nonlinear model due to a specific dynamical feature that is of interest. However, it should be noted that there have been studies where the sensitivities to model parameters can also be determined by using adjoint sensitivities.

Another important role that adjoints play is in an approach to determine causes of forecasts busts. If a forecast has predicted the wrong event and there are observations of the true event, then by taking the difference between these two events, it is possible to determine a sensitivity to a specific sets of model variables at specific times which could have caused the bad forecast. The usefulness of this is that it enables modelers and data assimilation scientists to determine if there is an error in the numerical model or if there is a mis-characterization of the probabilistic behavior of that variable at that location in the data assimilation scheme that led to the incorrect initial conditions being supposedly optimal.

13.6 Singular Vectors

We start by denoting a general autonomous system of a state variable $\boldsymbol{x}$, whose evolution equation can formally be written as

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{M}(\boldsymbol{x}). \tag{13.116}$$

We shall denote the time integration of (13.116) from time t_0 to t as $\boldsymbol{x}(t)$, which generates a trajectory from an initial point, $\boldsymbol{x}_0$, to $\boldsymbol{x}_1 = \boldsymbol{x}(t)$. The time evolution of a small perturbation $\delta\boldsymbol{x}$ around the time evolving trajectory $\boldsymbol{x}$ can be described, as we saw earlier, through the linearization of (13.116), which is denoted by

$$\frac{\delta\boldsymbol{x}}{\partial t} = \mathbf{M}\delta\boldsymbol{x}, \tag{13.117}$$

where $\mathbf{M} = \left.\dfrac{\partial\mathcal{M}(\boldsymbol{x})}{\partial\boldsymbol{x}}\right|_{\boldsymbol{x}(t)}$.

We now introduce the integral forward propagator of the dynamical system $\mathbf{L}(t, t_0)$ linearized about the nonlinear trajectory $\boldsymbol{x}(t)$; this enables us to define the evolution of the perturbation from its initial value at time t_0 to what is referred to as the *optimization time* t as

$$\delta\boldsymbol{x}(t) = \mathbf{L}(t, t_0)\,\delta\boldsymbol{x}(t_0). \tag{13.118}$$

If we now denote the linear vector space of the perturbations at $\boldsymbol{x}_0$ by Λ_0, and the linear vector space at $\boldsymbol{x}_1$ by Λ_1, then $\{\mathbf{L} : \Lambda_0 \to \Lambda_1\}$.

If we now consider two perturbations, $\delta\boldsymbol{x}$ and $\delta\boldsymbol{y}$, and if we have a Hermitian matrix, $\mathbf{E}$, then we define the inner product $(\cdot,\cdot)_{\mathbf{E}}$ as

$$(\delta\boldsymbol{x}, \boldsymbol{y})_{\mathbf{E}} \equiv \langle\delta\boldsymbol{x}, \mathbf{E}\boldsymbol{y}\rangle, \tag{13.119}$$

on the tangent space Λ_0, where $\langle\cdot,\cdot\rangle$ identifies the canonical Euclidean scale product, which is given by

$$\langle\delta\boldsymbol{x}, \delta\boldsymbol{y}\rangle \equiv \sum_{i=1}^{N} \delta x_i \delta y_i. \tag{13.120}$$

The next important step in the derivation of the singular vectors is to define a norm that is associated with the inner product in (13.119); this norm is denoted by $\|\cdot\|_{\mathbf{E}}^2$ and is defined as

$$\|\delta \boldsymbol{x}\|_{\mathbf{E}}^2 = (\delta \boldsymbol{x}, \delta \boldsymbol{x})_{\mathbf{E}} = \langle \delta \boldsymbol{x}, \mathbf{E}\delta \boldsymbol{x} \rangle . \tag{13.121}$$

Next we denote the adjoint of $\mathbf{L}$ with respect to the inner product $(\cdot, \cdot)$ as $\widehat{\mathbf{L}}^{\mathbf{E}}$, which is defined as

$$\left(\widehat{\mathbf{L}}^{\mathbf{E}} \delta \boldsymbol{x}, \boldsymbol{y}\right) = (\delta \boldsymbol{x}, \mathbf{L}\delta \boldsymbol{y})_{\mathbf{E}} . \tag{13.122}$$

An important property that we need to take note of here is that the adjoint of $\mathbf{L}$ with respect to the inner product defined by $\mathbf{E}$ can be written in terms of the adjoint $\widehat{\mathbf{L}}$, defined with respect to the canonical Euclidean scalar product as

$$\widehat{\mathbf{L}}^{\mathbf{E}} = \mathbf{E}^{-1}\widehat{\mathbf{L}}\mathbf{E}. \tag{13.123}$$

From (13.118) and (13.123) it follows that the squared norm of a perturbation $\delta \boldsymbol{x}$ at time t is given by

$$\|\delta \boldsymbol{x}\,(t)\,\|_{\mathbf{E}}^2 = \left(\delta \boldsymbol{x}\,(t_0)\,, \widehat{\mathbf{L}}^{\mathbf{E}}\mathbf{L}\delta \boldsymbol{x}\,(t_0)\right)_{\mathbf{E}} . \tag{13.124}$$

Equation (13.124) shows that the problem of finding the phase space directions $\delta \boldsymbol{x}$ for which $\frac{\|\delta \boldsymbol{x}(t)\|_{\mathbf{E}}^2}{\|\delta \boldsymbol{x}(t_0)\|_{\mathbf{E}}^2}$ is maximum can be reduced to the search of the eigenvectors $\boldsymbol{v}_i\,(t_0)$, from the eigenvalue problem

$$\widehat{\mathbf{L}}^{\mathbf{E}}\mathbf{L}\boldsymbol{v}_i\,(t_0) = \sigma_i^2 \boldsymbol{v}_i\,(t_0)\,, \tag{13.125}$$

with the largest eigenvalue denoted by σ_i^2.

The square roots of the eigenvalues σ_i are referred to as the **singular values** and the eigenvectors $\boldsymbol{v}_i\,(t_0)$ are the right **singular vectors** of $\mathbf{L}$ with respect to the inner product $\mathbf{E}$. The singular vectors with the largest singular values identify the directions that are characterized by the maximum growth. The time interval $t - t_0$ is referred to as the **optimization time interval**.

A property of $\mathbf{L}$ is that it is not normal; however, the operator $\widehat{\mathbf{L}}^{\mathbf{E}}\mathbf{L}$ is normal, which implies that its eigenvectors $\boldsymbol{v}_i\,(t_0)$ can be chosen to form a complete orthonormal basis in the Nth dimensional tangent space of the perturbation at $\boldsymbol{x}_0$, but also that the eigenvalues are real and positive.

If we now consider the implications of this derivation, then at the optimization time t we have that the singular vectors evolve to

$$\boldsymbol{v}_i\,(t) = \mathbf{L}\,(t, t_0)\,\boldsymbol{v}_i\,(t_0)\,, \tag{13.126}$$

which then satisfy the eigenvector problem

$$\mathbf{L}\widehat{\mathbf{L}}^{\mathbf{E}}\boldsymbol{v}_i\,(t) = \sigma_i^2 \boldsymbol{v}_i\,(t_0)\,. \tag{13.127}$$

We can now relate the eigenvalues with the norm associated with the inner product through

$$\|\boldsymbol{v}_i\,(t)\,\|_{\mathbf{E}}^2 = \sigma_i^2. \tag{13.128}$$

Since any perturbation scaled as $\frac{\delta \boldsymbol{x}(t)}{\|\delta \boldsymbol{x}(t_0)\|_{\mathbf{E}}}$ can be written as a linear combination of the singular vectors $\boldsymbol{v}_i$, it follows that

$$\max_{\|\delta \boldsymbol{x}(t_0)\|_{\mathbf{E}}} \left(\frac{\|\delta \boldsymbol{x}\,(t)\,\|_{\mathbf{E}}}{\|\delta \boldsymbol{x}\,(t_0)\,\|_{\mathbf{E}}} \right) = \sigma_i \,. \tag{13.129}$$

Therefore, the maximum growth as measured by the norm $\|\cdot\|_{\mathbf{E}}$ is associated with the dominant singular vector $\boldsymbol{v}_i$. However, we should note that given the tangent forward propagator $\mathbf{L}$, then from (13.125) it implies that the singular vectors' characteristics depend strongly on the inner product definition and to the specification of the optimization time interval.

As an example of the effects that different choices have for the metric which the inner product is defines with respect to, see [325] where the authors consider streamfunction variance, entropy, kinetic energy, and total energy norms. The European Center for Medium Range Forecasting use the singular vectors of the linear version of their model that have been computed to maximize the total energy norm over a 48-hour time interval. Given these singular vectors, they then use them to initialize their ensemble prediction system which is to sample the directions where the maximum energy growth of the 48-hour period has been identified.

The technique described above identifies the directions of the nonlinear model that could have the maximum growth, but also identifies where the model at the time of calculation has the most sensitivity. Now we consider how we can use the adjoint model to identify observational sensitivities.

13.6.1 Observational Impact

We have seen earlier in this chapter we can use the adjoint of the numerical forecast model to identify regions of the geophysical system where forecast error growth in numerical forecast models is maximally sensitive to the errors in the initial conditions, in advance. Given these identified sensitive areas, we could add extra observing vehicles to reduce the forecast error there as we could assimilate more observations to constrain the forecast errors [21].

As stated in Baker and Daley [21], these adjoint techniques do not take into account the characteristics of the data assimilation system used in the analysis of these **targeted observations**. An important point is that these model adjoint schemes do not take in to consideration the interaction of the background field, interactions with other observations, and the background and observation error characteristics. It is therefore possible that we could have a missampling, conflict with other observations, and inefficient use of aircraft and expendables.

Baker and Daley [21] develop an alternative manner in which to design a target observation strategy through considering the adjoint of a simplified data assimilation system. We now summarize their technique for the adjoint of the data assimilation system: let the 3D analysis equation be given by

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{K}\left(\mathbf{y} - \mathbf{h}\left(\mathbf{x}^b\right)\right), \tag{13.130}$$

where $\mathbf{x}^a$ is referred to as the analysis state, and $\mathbf{x}^b$ is the background state; this could be a wide range of possibilities, but the most common form for the background state is the forecast from the previous analysis cycle, but it can also be a climatology, $\mathbf{y}$ represents the observations and $\mathbf{h}$ is the possible nonlinear observation operator, $\mathbf{K}$ is referred to as the gain matrix, and we shall introduce all of these terms in the next few chapters.

We now linearize the observation operator in (13.130) such that our analysis step is now

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^b\right), \tag{13.131}$$

where $\mathbf{H}$ is the linearized observation operator about the background state. The gain matrix can be shown to be

$$\mathbf{K} = \mathbf{P}_b\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_b\mathbf{H}^T + \mathbf{R}\right)^{-1}, \tag{13.132}$$

where $\mathbf{P}_b$ is referred to as the background error covariance matrix, and $\mathbf{R}$ is the observations error covariance matrix. It is possible to write (13.131) as

$$\mathbf{x}^a = (\mathbf{I} - \mathbf{K}\mathbf{H})\,\mathbf{x}^b + \boldsymbol{K}\mathbf{y}. \tag{13.133}$$

We now require the sensitivity of the analysis state to the observations, $\frac{\partial \mathbf{x}^a}{\partial \mathbf{y}}$ and the sensitivity to the background state $\frac{\partial \mathbf{x}^a}{\partial \mathbf{x}^b}$. It is stated in [21] that the derivative of a vector with respect to another vector is a matrix, and as such the two expressions that we seek provide the following matrices:

$$\frac{\partial \mathbf{x}^a}{\partial \mathbf{y}} = \mathbf{K}^T, \tag{13.134a}$$

$$\frac{\partial \mathbf{x}^a}{\partial \mathbf{x}^b} = \mathbf{I} - \mathbf{H}^T\mathbf{K}^T. \tag{13.134b}$$

The next step is to relate (13.134a) and (13.134b) to the adaptive targeting problem. As with the model adjoint approach, we define a forecast verification domain that is the subset of the total domain and a cost function J, that is a scale measure of some forecast quantity of interest over the forecast verification domain. The scale J is referred to as the **forecast aspect** in [21]. The gradient of J with respect to the initial conditions for the forecast is $\frac{\partial J}{\partial \mathbf{x}^a}$. This vector is referred to as the **analysis sensitivity** vector.

The goal of the section is to determine the sensitivity of the forecast aspect to the observations, $\frac{\partial J}{\partial \mathbf{y}}$, which is a vector of length of the number of observations while the analysis sensitivity vector is of length of the model state vector, and this vector is referred to as the **observation sensitivity vector**. Another measure that is of interest is that which determines the sensitivity of the forecast aspect to the background field $\frac{\partial J}{\partial \mathbf{x}^b}$; this vector is referred to as the **background sensitivity vector**. Through applying the chain rule, it is possible to show that these two sensitivity measures are given by

$$\frac{\partial J}{\partial \mathbf{y}} = \frac{\partial \mathbf{x}^a}{\partial \mathbf{y}}\frac{\partial J}{\partial \mathbf{x}^a} = \mathbf{K}^T\frac{\partial J}{\partial \mathbf{x}^a}, \tag{13.135a}$$

$$\frac{\partial J}{\partial \mathbf{x}^b} = \frac{\partial \mathbf{x}^a}{\partial \mathbf{x}^b}\frac{\partial J}{\partial \mathbf{x}^a} = \left(\mathbf{I} - \mathbf{H}^T\mathbf{K}^T\right)\frac{\partial J}{\partial \mathbf{x}^a}. \tag{13.135b}$$

Substituting the expression for the gain matrix from (13.132), we can write the sensitivity measure above as

$$\frac{\partial J}{\partial \mathbf{y}} = \left(\mathbf{H}\mathbf{P}_b\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\mathbf{P}_b\frac{\partial J}{\partial \mathbf{x}^a}, \tag{13.136a}$$

$$\frac{\partial J}{\partial \mathbf{x}^b} = \left(\mathbf{I} - \mathbf{H}^T\left(\mathbf{H}\mathbf{P}_b\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\mathbf{P}_b\right)\frac{\partial J}{\partial \mathbf{x}^a}. \tag{13.136b}$$

It is possible to define the **analysis space projection of the observation sensitivity vector** as

$$\mathbf{H}^T\frac{\partial J}{\partial \mathbf{y}} = \frac{\partial J}{\partial \mathbf{x}^a} - \frac{\partial J}{\partial \mathbf{x}^b}. \tag{13.137}$$

This now means that we have a method of calculating the sensitivity of the forecast aspect with respect to the observations $\frac{\partial J}{\partial \mathbf{y}}$ and the gradient of the forecast aspect with respect to the background field $\frac{\partial J}{\partial \mathbf{x}^b}$.

To implement the sensitivity measure just described, we require the background and observational error covariances along with the position and types of the observations, as well as the linearized observation operator and its adjoint. The adjoint of the observation operator enables us to transform variables between observation and grid space.

An important application of the sensitivity measure above is presented in Langland and Baker [237]. Here they define the forecast aspect measure to be scalar forecast error norm, given by

$$e_f = \left\langle \left(\mathbf{x}^f - \mathbf{x}^t\right), \mathbf{C}\left(\mathbf{x}^f - \mathbf{x}^t\right)\right\rangle, \tag{13.138}$$

where $\mathbf{x}^t$ is the verifying analysis, and $\mathbf{C}$ is a matrix of energy weighting coefficients that represents dry total energy [358]. The cost function to perform the adjoint sensitivity calculation with respect to is

$$J = \frac{1}{2}e_f, \tag{13.139}$$

where the starting conditions for the adjoint integration, valid at the forecast time is

$$\frac{\partial J}{\partial \mathbf{x}^f} = \mathbf{C}\left(\mathbf{x}^f - \mathbf{x}^t\right). \tag{13.140}$$

A difference in the approach in [237] compared to [21] is that the authors perform a single integration of the forecast model adjoint, providing a 3D sensitivity vector with respect to the initial conditions of the forecast trajectory

$$\frac{\partial J}{\partial \mathbf{x}^a} = \mathbf{L}^T\frac{\partial J}{\partial \mathbf{x}^f}, \tag{13.141}$$

where $\mathbf{L}^T$ is the adjoint of the numerical model. The sensitivity gradient $\frac{\partial J}{\partial \mathbf{x}^a}$ can be used to estimate how J is changed by adding a small perturbation to $\mathbf{x}^a$.

We shall see in Chapter 18 that the US Navy's data assimilation system is performed in observation space, which is the data assimilation system that [237] uses for its sensitivity study, and as such its analysis state can be written as

$$\mathbf{x}^a - \mathbf{x}^b = \mathbf{P}_b\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_b\mathbf{H}^T + \mathbf{R}\right)^{-1}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^b\right). \tag{13.142}$$

We shall prove this expression above in many of the chapters on data assimilation to come.

The next step is to extend the initial conditions sensitivity gradient from grid-point space from (13.141) into observation space through the adjoint of the data assimilation system as

$$\frac{\partial J}{\partial \mathbf{y}} = \left(\mathbf{H} + \mathbf{P}_b \mathbf{H}^T + \mathbf{R}\right)^{-1} \mathbf{H}\mathbf{P}_b \frac{\partial J}{\partial \mathbf{x}^a}. \tag{13.143}$$

We can apply the theory described above if we consider the difference between two different forecasts, valid at the same time, but have been started from two different analysis steps of the data assimilation scheme, so that the difference between the two forecasts is the effect of assimilating the observations at the later assimilation time.

We let $\mathbf{e}_g$ be the forecast error from the assimilation scheme started one cycle before the current assimilation cycle, and let $\mathbf{e}_f$ be the forecast error for the forecast started from the current assimilation cycle. We have a copy of the schematic of this setup from [237] in Fig. 13.8. Therefore, the difference between the forecast errors $\Delta \mathbf{e}_f^g = \mathbf{e}_f - \mathbf{e}_g$ is due to the assimilation of the observations. Therefore, using observation sensitivity gradients, $\delta \mathbf{e}_f^g$ can be approximated as

$$\delta \mathbf{e}_f^g = \left\langle (\mathbf{y} - \mathbf{H}\mathbf{x}_b), \frac{\partial J_f^g}{\partial \mathbf{y}} \right\rangle, \tag{13.144}$$

where

$$\begin{aligned} J_f &= \frac{1}{2} \left\langle \left(\mathbf{x}_f - \mathbf{x}_t\right), \mathbf{C}\left(\mathbf{x}_f - \mathbf{x}_t\right) \right\rangle, \\ J_g &= \frac{1}{2} \left\langle \left(\mathbf{x}_g - \mathbf{x}_t\right), \mathbf{C}\left(\mathbf{x}_g - \mathbf{x}_t\right) \right\rangle, \\ \Rightarrow \frac{\partial J_f^g}{\partial \mathbf{y}} &= \boldsymbol{K}^T \left(\frac{\partial J_f}{\partial \mathbf{x}^a} + \frac{\partial J_g}{\partial \mathbf{x}^b} \right). \end{aligned}$$

A full derivation and explanation of the expressions above can be found in [237].

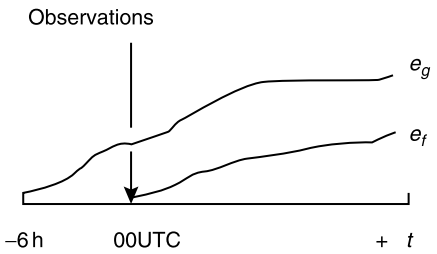


FIGURE 13.8

Copy of the schematic of the errors from Rolf H. Langland & Nancy L. Baker (2004) Estimation of observation impact using the NRL atmospheric variational data assimilation adjoint system, Tellus A: Dynamic Meteorology and Oceanography, 56:3, 189-201, DOI: 10.3402/tellusa.v56i3.14413. https://creativecommons.org/licenses/by/4.0/.

13.7 Summary

In this chapter we have shown how to derive the tangent linear model and the adjoint of nonlinear models. These two numerical models play a vital part in geophysical prediction in the form of studying error growth as well as in variational data assimilation. We have presented the derivation of the tangent linear model for the Lorenz 1963 model, the Eady model, for semi-Lagrangian advection with the cubic Lagrange interpolation scheme, as well as for spectral methods. We have introduced an alternative to the tangent linear model, called the perturbation forecast model, where instead of linearizing the numerical source code for the nonlinear model, we linearize the nonlinear continuous model and then discretize this model. There is much debate about which of these approaches is more efficient, accurate, but also extendable to more complex systems as the resolution of certain numerical models increase. Another way of posing this question is: is it better to discretize then linearize, or linearize and then discretize? Different people have different answers to this question and we shall not favor one over the other.

We have seen that adjoints play an important part in the singular vector calculations which enable modelers to determine regions of the numerical model where there are fast growths of the error that could lead to problems for the data assimilation schemes. We have also introduced the observational impact measure using adjoints that can help to identify which sets of observations appear to have the best impact on a system, but also through data denial experiments we can see which sequence of observations provide the most information.

Another important feature of adjoints that has been presented is that it is possible to use them to determine the impact of an assimilation system. This is important when performing target observations experiments, but it is also important for deciding if a satellite is coming to the end of its lifetime, and there is no replacement that is going to be ready in time, or we need to justify that the satellite should be replaced, then the measure in the last section is a good tool to help determine the impact, both positive or negative, of having, or not having, those data.

This is the last chapter that involves some form of numerical modeling, which is the first part of setting up a data assimilation system. We now move on to observations of geophysical systems, which is the other key component that we need before we introduce the different forms of data assimilation algorithms.

This page intentionally left blank

CHAPTER

Observations 14

Contents

Over the last six chapters, we have introduced different theories and techniques for the numerical modeling of ordinary and partial differential equations that could be applied to the governing equations of different aspects of the geophysical system. Once the numerical model has been determined, then the observations that we would like to assimilate to control the errors in the numerical model have to be decided upon. Numerical modeling is just one part of data assimilation, and as such we now consider the second component, which is the observations.

Since the introduction of satellites in the 1960s, the number of observations available for different operational and research centers that employ a form of data assimilation has increased dramatically. However, it is often the case that the sensor data from the satellites, where the process of the satellite obtaining data about different geophysical systems is referred to as **remote sensing**, do not directly

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00018-8

observe the variables that are the drivers of the numerical model; there are known relationships/approximations between different geophysical variables and **radiances** or **brightness temperatures**, so that we can invert the relationship between the satellite observations and the geophysical variables.

In this chapter we shall introduce the concept of direct and indirect observations, as well as the theory that links satellite observations and geophysical variables. We shall look at the link between radars and different geophysical variables, as well as the Global Positioning System's (GPS) occultation observations that are related to humidity. We shall also present a summary of the geophysical variables that can be remotely sensed by microwaves. Finally, we shall finish this chapter with a look at some different methods for **quality control and bias correcting** the observations.

14.1 Conventional Observations

In this section we summarize different sets of observations that are taken from the ground for many different geophysical variables. We shall consider observations from many different centers, but the program that has the most different sets of data available for different atmospheric and soil variables is the Atmospheric Radiation Measurement (ARM). The ARM program has multiple sites around the United States that collect geophysical observations from multiple sensors.

Before the introduction of satellites, most geophysical observations were obtained through ground, under water, or airborne forms. These observations are still collected today and play a very important role in operational data assimilation systems. They could be direct observations, where you are observing the geophysical variable itself, subject to an interpolation in space and/or time, or indirect, as just mentioned above the sensors on the satellites, but these observations are collected from platforms within the atmosphere, ocean, or on or near the surface. The first such atmospheric device we consider here is the **radiosonde**.

14.1.1 Radiosondes

Radiosondes are battery-powered telemetry instrument packages that are carried into the atmosphere typically by a weather balloon; they measure altitude, pressure, temperature, relative humidity, wind (both speed and direction), and cosmic ray readings at high altitudes. A class of radiosonde whose position is tracked as it ascends in the atmosphere to give wind speed and direction is referred to as **rawindsonde**, which is an abbreviation for radar wind sonde. Another class of radiosondes are the ones that are released from airplanes and fall rather than being carried by weather balloons. This class of radiosondes are referred to as **dropsondes**. Radiosondes play a vital part in most forms of operational atmospheric data assimilation.

An important feature to note about the different forms of radiosondes is that the observation may not occur at a model, or an *analysis*, grid point, and as such there will have to be some interpolation from the model to the location of the radiosonde observation as we just mentioned.

14.1.2 Microwave Radiometer

There are many different versions of the radiometer; the first we consider here is the high-frequency microwave radiometer (MWRHF) which provides time series measurements of brightness temperature

from two channels centered at 90 and 150 Hz. These two channels are sensitive to the presence of liquid water and precipitable water vapor. This sensor can be found at the ARM sites and it measures atmospheric moisture, pressure, temperature, and precipitation.

The three-band radiometer provides time series measurements of brightness temperature from three channels centered at 23.834, 30, and 89 GHz. These three channels are sensitive to the presence of liquid water and precipitable water vapor. The observations that are produced by this sensor are liquid water path and precipitable water.

14.1.3 Infrared Sky Imager

The infrared sky imager is an automatic continuously operating digital imaging and software system that captures hemispheric sky digital images, along with time series of retrievals of fractional sky cover, but from a data assimilation perspective, this sensor produces cloud fraction observations.

14.1.4 Micropulse Lidar

The micropulse lidar is a ground-based optical remote sensing system that has been designed to determine the altitudes of clouds overhead. Therefore, the observation that this sensor produces is cloud base height.

14.1.5 Photometer

The first photometer we consider here is the particle soot absorption photometer, which is a device that collects aerosol particles on a substrate and measures the change in light transmission relative to a reference filter. The observation that this sensor provides is aerosol absorption.

The second photometer is the continuous light absorption photometer; this operates on the same premise as for the particle soot absorption photometer, and is able to produce optical properties of the aerosols as an observation.

14.1.6 SNOTEL

SNOTEL stands for SNOwpack TELemetry and is an extensive, automated system to collect snowpack and related climatic data in the Western United States. It evolved from a Congressional mandate in the 1930s to the Natural Resource Conservation Service (NCRS) to measure snowpack in the mountains of the West and to forecast water supply.

There are two types of SNOTEL stations that are in use in the USA. The first is the basic station, which has a pressure-sensing snow pillow, storage precipitation gage, and an air temperature sensor. From the NRCS it is possible to obtain daily precipitation accumulation by water year, historic and by historic monthly accumulation. The daily maximum, minimum, and average temperature by current water year and historic are also available.

With respect to snow, the daily snow water equivalent (SWE) by current water year, and by historic year are available as well.

14.1.7 SCAN

SCAN stands for the **Soil Climate Analysis Network** and began as a pilot soil moisture and soil temperature project at the NRCS in 1991. The SCAN system focuses on agricultural areas of the United States and comprises of over 200 stations. Each of the stations in SCAN measures wind speed and direction, liquid precipitation, soil moisture, and temperature sensors at different depths beneath the surface; they have a relative humidity and air temperature sensor, and also a solar radiation sensor.

14.1.8 Airborne Observations

When certain aircraft are flying, they are able to provide different meteorological data through the Aircraft Communication Addressing and Reporting System (ACARS) and the Aircraft Meteorological Data Relay (AMDAR), which is part of the World Meteorological Organization (WMO). AMDAR reports are taken every 7 minutes at cruising altitude, where are extra reports at wind maxima. During the plane's ascent the sensors report every 10 hPa intervals vertically of the first 100 hPa in the lower atmosphere and every 50 hPa above that layer, with the reverse happening on the descent. The AMDAR system thus provides data about every 70–100 km along the flight path, but then offer higher resolution in the vicinity of airports. Flight-based data are used in operational data assimilation systems. For more information about the initial impacts of these observations on the ECMWF data assimilation system, see [55].

14.1.9 Ocean

In this section we summarize different observing platforms that are collecting data that could and in most cases are assimilated into an operational ocean prediction data assimilation system.

Buoys

The Global Tropical Moored Buoy Array Program is an international effort to provide data in real-time for climate research and forecasting. Major components include the Tropical Atmosphere Ocean (TAO)/ Triangle Trans-Ocean Buoy Network (TRITON) array in the Pacific, Prediction and Research Moored Array in the Tropical Atlantic (PIRATA) in the Atlantic, and Research Moored Array for African-Asian-Australian Monsoon Analysis and Prediction (RAMA) in the Indian Ocean. The buoys are platforms for instruments to measure solar and thermal radiation as well as air-sea exchange of carbon dioxide in the tropics.

Another type of mooring that provides usual observations of the ocean and the atmosphere is the Autonomous Temperature Line Acquisition System (ATLAS). The buoys associated with ATLAS consists of a conducting cable below the surface of the ocean that measures sea temperature, conductivity, and pressure at various levels below the ocean surface. There is also a sensor at the surface that measures surface temperature, conductivity, and pressure up to a meter below the surface. Above the surface there are sensors to measure air temperature and relative humidity, along with short-wave radiation, a wind sensor, and a rain gauge.

A third set of buoys that are important for both ocean and atmospheric data assimilation are the **drifting buoys**. These are part of the global drifter program that is an array of about 1250 drifting buoys, again at the time of writing, that are distributed globally in the main oceans and the Mediterranean Sea.

The drifter buoys have instruments to measure sea surface temperature, barometric pressure, wind, ocean color, and salinity, as well as subsurface ocean temperatures.

Another set of buoys that are used for ocean data assimilation are the tsunameter buoys. These buoys are part of the Deep-ocean Assessment and Reporting of Tsunamis (DART) mooring system, and they provide not only observations of the movement of the surface of the ocean and water pressure, but also important meteorological and ocean observations of wind, barometric pressure, sea surface temperature and conductivity, air temperature, and relative humidity.

Expendable bathythermographs

Expendable bathythermographs (XBT) are temperature probes that are dropped into the ocean from what are quite often referred to as *ships of opportunity*, where these are commercial carriers that transit scientifically important trans-oceanic routes. The XBTs are able to measure temperature of the upper kilometer of the ocean.

Argo floats

Argo is an international collaboration that collects high-quality temperature and salinity profiles from the upper 200 m of the ice-free global ocean and currents from intermediate depths. The Argo floats are typically submerged at around 200 m for approximately 10 days. At this time the floats starts to ascend toward the ocean surface, which takes approximately 6 hours while measuring temperature and salinity.

Arctic buoys

The international Arctic buoys program is a network of automatic buoys that measure sea level pressure, surface air temperature, and the ice motion throughout the Arctic Ocean. The buoys are also able to record snow depth, sea ice thickness, sea ice temperature, and ocean temperature and salinities.

Marine mammals

Temperature and salinity profiles are being obtained from instrumented marine mammals in near-real time. The mammals sample to depths of up to 2000 m in high-latitude regions where there are few other in situ profile observations. One example is the southern elephant seals that swim around Antarctica. The mammals provide data in regions where there are not many other observing platforms, and provide vital information about the ocean in these regions. For an example of how the assimilation of mammal data has had an impact on ocean data assimilation, see [58].

14.1.10 Radar

Radar plays a crucial role in severe storm predication from the radars that are based on the Great Plains of the United States where they provide observations of supercells, development of snow storms and blizzards throughout the Rockies and the Front Range in Colorado, track hurricanes as well as rain/hail intensity. Radars are becoming widespread in more countries, and are important not just to meteorology. High-frequency radar can be used to derive the surface current of the ocean [116]. Another important property of radar observations is that they have high spatial and temporal resolution and can be mobile and non-mobile on the ground, as well as being on aircraft.

Doppler radar data consists of the radial velocity and the reflectivity, where the radial velocity contains information about the vertical motion, which can be important for convective initiation and forecasting, and can also be used to determine surface currents on the ocean as we mentioned above. The reflectivity observations provide measurements of what are referred to as **precipitation hydrometeors**, which are rain, snow, drizzle, hail, ice pellets, graupel, and different combinations of liquid to solid (frozen) states of these.

As with all indirect observations, we have to develop the **observation (forward) operator** to be able to find a relationship between the observed quantity and the variables of either the model, or a transform of the model. As a result of the vertical velocity being an important component of the radial velocity observations, it is necessary to find a method to enable a data assimilation scheme to assimilate an increment to the vertical wind, w. In [484] the authors introduce a linearized version of Richardson's equation, via a base state and a small increment to the pressure field, p, and the three-dimensional wind vector $\boldsymbol{u}^T \equiv (u, v, w)$, which results in a new balanced equation of the form

$$\begin{aligned}\gamma\overline{p}\frac{\partial w'}{\partial z} = &-\gamma p'\frac{\partial \overline{w}}{\partial z} - \gamma\overline{p}\nabla\cdot\boldsymbol{u}_h' - \gamma p'\nabla\cdot\overline{\boldsymbol{u}}_h - \overline{\boldsymbol{u}}_h\cdot\nabla p' - \boldsymbol{u}_h'\cdot\nabla\overline{p} \\ &+ g\int_z^\infty \nabla\cdot\left(\overline{\rho}\boldsymbol{u}_h'\right)dz + g\int_z^\infty \nabla\cdot\left(\rho'\overline{\boldsymbol{u}}_h\right)dz,\end{aligned} \tag{14.1}$$

where $\boldsymbol{u}_h$ is the horizontal wind vector, γ is the ratio of specific heat capacities of air at constant pressure-volume, ρ is the density, z is the height, and g is the acceleration due to gravity.

The observation operator for the radial velocity is given by

$$\boldsymbol{u}_r = u\frac{x - x_i}{r_i} + v\frac{y - y_i}{r_i} + (w - \mathbf{v}_T)\frac{z - z_i}{r_i}, \tag{14.2}$$

where (x, y, z) are the location of the radar, (x_i, y_i, z_i) are the location of the radar observations, r_i is the distance between the radar and the observations, and $\mathbf{v}_T$ is the terminal velocity, which is defined as

$$\mathbf{v}_T = 5.40aq_r^{0.125}, \tag{14.3}$$

where q_r is the mixing ratio of rainwater and a is a *correction factor* that is defined as

$$a = \left(\frac{p_0}{\overline{p}}\right)^{0.4}, \tag{14.4}$$

where $\overline{p}$ is the base state pressure and p_0 is the pressure at the ground.

For the observation operator associated with the doppler radar reflectivity, there are many different forms that this operator can take. This is due to the hydrometeors that could be assimilated, and/or predicted. If you are only considering rainwater as a predicting variable, then in [484] the authors define the observation operator for reflectivity as

$$Z = 43.1 + 17.5\log\left(\rho q_r\right), \tag{14.5}$$

where the logarithm is to base 10.

As we mentioned above, (14.2) and (14.5) are for the rainwater mixing ratio case. In [59] the authors introduce the equivalent to (14.2) and (14.5) for snow. The terminal velocity operator for the mixing

ratio for snow becomes

$$v_{Ts} = 0.97 \left(\frac{p_0}{\overline{p}} \right)^{0.4} (\rho q_s)^{0.125}, \tag{14.6}$$

where q_s is the mixing ratio for snow. The equation for the observation operator for the radar reflectivity with respect to snow is given as

$$Z_s = 31.1 + 17.5 \log (\rho q_s), \tag{14.7}$$

where the logarithm is again to base 10. In [59] the authors provide a detailed mathematical description of the tangent linear and the adjoints of these observation operators, which are important in variational and ensemble-based data assimilation schemes.

Polarimetric radars are radars that emit radiation in both the horizontal and the vertical, and can be used to observe precipitation type and rates; see [446] for an example of this type of observation from this type of radar. There are also wind profilers radars that transmit pulses of electromagnetic radiation vertically and in at least two slightly off vertical directions in order to resolve the three-dimensional vector winds. The backscatter that the profilers receive return from atmospheric features such as turbulence, clouds, and precipitation as well as non-atmospheric features such as insects, birds, trees, airplanes, and radio frequency interference. Assimilation of wind profiler observations can play an important part with respect to wind farm production of electricity [98].

14.2 Remote Sensing

As we mentioned at the beginning of this chapter, with the introduction of satellites in the 1960s, the number of observations that are available for different geophysical modeling and assimilation has drastically increased. However, for most sensors on the satellites their observations are not directly of the variable of concern, but there exists relationships/approximations between the radiance/brightness temperature and the geophysical variable. In this section we shall briefly introduce the different types of space-bound remote sensing sensors that are available. We should note that some of the space-bound sensors are also available on the ground as well, as we saw with the radiometers in the ground-based section. However, in order to be able to use the observations from these non-conventional observations, we have to understand how to infer the relationships from the radiance/brightness temperature and the geophysical variables. This is referred to as **radiative transfer theory** and we shall provide a brief overview now.

14.2.1 Radiative Transfer Modeling

The starting point for radiative transfer theory is to consider the electromagnetic spectrum. The spectrum consists of gamma rays, X-rays, ultraviolet, visible, infrared, microwave, and finally radio waves.

The spectrum is the range of all of these types of electromagnetic radiation. The electromagnetic radiation can be expressed in terms of energy, wavelength, or frequency. Frequency is measures in cycles per second, or **Hertz**, wavelength is measured in meters, while energy is measured in electron volts. Given these measures of the radiation, we have that spectrum ranges from short wavelengths with

high frequency and higher energy, gamma and X-rays, to the longer wavelengths, lower frequency, and lower energy waves (radio).

Two sets of electromagnetic waves that are most frequently used in data assimilation are microwave and infrared waves. The reason for this is that there are known interactions of these forms of radiation with geophysical variables that are of interest to us in different forms of geophysical-based data assimilation, and of course by association, forecasting.

The detectable affect of microwave radiation on geophysical variables is that it rotates the molecules, which results in a change in the dipole.

For the infrared radiation there are three different process that can be detected. The first is vibration of the molecules with results in symmetric stretching, asymmetric stretching, or a bending motion.

It is the change in a molecule's dipole moment, which could be either electric or magnetic, as it rotates, or vibrates, that allows the molecule to interact with the radiation. The energies at which molecules interact with radiation are determined by properties such as their shape, which determines the rotational inertia, and bond strength, which determines the elasticity of stretching and bending moments. The rotational and vibrational energies of the molecules have been quantized. This means that molecules interact with radiation at well-defined frequencies.

Therefore, if we have a measurement of the spectra, then we have information about the state of the atmosphere, ocean, and the surface; these pieces of information can be thought of as answers to the following three questions: (1) How much of molecule X is there? (2) What is the temperature, at pressure p? and finally (3) What is the surface temperature? To be able to extract information from radiance measurements, we are solving an inverse problem; we shall go into more detail about what these are in the next chapter. However, the first priority here is to be able to simulate the **transfer** of radiation through the atmosphere.

Before we progress, we introduce the following terms and laws that are fundamental to radiative transfer theory. The first two are those of **absorption**, where the radiance decreases, and **emission**, where the radiance increases. Given these two simple definitions, we then have the following two laws:

Definition 14.1 (Kirchoff's Law). Under the conditions of local thermodynamic equilibrium, then the absorptivity, a, of a medium is equal to its emissivity ε.

Definition 14.2 (Beer-Bouguer-Lambert's Law). The absorption (extinction) process is linear, independent of the radiation intensity and amount of matter, provided that the physical state is held constant.

The second law above has many different names that are different combinations of the three surnames given above.

Given these two laws, we consider the interaction of radiation and matter. From Lambert's Law, the change of radiance, I, along a path, ds, to extinction is proportional to the amount of matter in the path; mathematically this is written as

$$dI_{abs} = -\beta_a I ds, \tag{14.8}$$

while we can write a similar expression for emissions as

$$dI_{emit} = \beta_a J ds, \tag{14.9}$$

where the β_a term is the volume absorption coefficient, which is

$$\beta_a = k_a \rho,$$

where k_a is the mass absorption coefficient, ρ is the density, and J is a source function, which is the Planck function, $B(T)$, under local thermodynamic equilibrium in a non-scattering medium.

If we now consider the total change in radiance, dI, due to the interaction of radiation and matter, then the associated equation is given by

$$dI = dI_{abs} + dI_{emit} = k_a \rho (B(T) - I) ds \Rightarrow \frac{dI}{k_a \rho ds} = B(T) - I, \tag{14.10}$$

where (14.10) is referred to as **Schwarzchild's equation**.

The next step is to define the **optical thickness**, τ, between two points along the path, s, and s_{toa} where toa stands for top of the atmosphere, which is given by

$$\tau (s \rightarrow s_{toa}) \int_{s}^{s_{toa}} k \rho ds' \Rightarrow d\tau (s \rightarrow s_{toa}) = -k \rho ds. \tag{14.11}$$

Substituting (14.11) into (14.10) and integrating the optical thickness, $O(s_{toa})$, to its value at the surface yields

$$I(O) = I\left(\tau_{sfc}\right) e^{-\tau_{sfc}} + \int_{0}^{\tau_{sfc}} B(\tau) e^{-\tau} d\tau. \tag{14.12}$$

Manipulation of (14.12) (see [211] for more details) leads to the **non-scattering radiative transfer equation**

$$I(p_{toa}) = B\left(T_{sfc}\right) \varepsilon_{sfc} t_{sfc} + \int_{p_{sfc}}^{p_{toa}} B(T(p)) \frac{d\tau(p)}{dp} dp, \tag{14.13}$$

where $t(p) = -\frac{\sec(\theta)}{g} e^{0\tau(p \rightarrow p_{toa})}$, and $\frac{d\tau(p)}{dp}$ is referred to as the weighting function.

We now consider the calculation of the transmittance, where we have to take into account the gaseous absorption algorithm via a regression model. We shall explain regression in the next chapter, involving frequency-dependent regression coefficients $c_{i,v}$, with atmospheric state predictors X_i to compute channel absorption coefficients that are given by

$$k_{a,v} = c_{0,v} + \sum_{i=1}^{N} c_{i,v} X_i. \tag{14.14}$$

Another feature that we consider is that of the radiative transfer equations when we have scattering. When we take into account the scattering, then the change in the intensity dI becomes

$$dI = DI_{ext} + dI_{emit} + dI_{scat}, \tag{14.15}$$

where $dI_{ext} - \beta_e I ds$ and $\beta_e = \beta_a + \beta_s$.

The dI_{scat} term takes into account radiation from any direction begin scattered into the sensor's field of view (FOV). Mathematically, this is expressed as

$$dI_{scat} = \frac{\beta_s}{4\pi} \int_{4\pi} p\left(\Omega', \Omega\right) I\left(\hat{\Omega}\right) d\omega' ds, \tag{14.16}$$

where $p\left(\Omega', \Omega\right)$ is the scattering phase function.

Given these relationships between certain geophysical variables and the radiation, we now introduce the characteristics of the satellite that the remote sensing sensors are housed upon.

14.2.2 Satellite Characteristics

Satellites orbit around the Earth, and other external bodies, as the result of the gravitational force, F_g, that is equivalent to the net centripetal force, F_c. When we assume that the orbits of the satellites are circular, then we have the two forces that we have to consider; these are

$$F_c = \frac{m_s v_s^2}{r_e + r_s}, \quad F_g = G\frac{m_s m_e}{(r_e + r_s)^2}, \tag{14.17}$$

where m_s is the mass of the satellite, r_e is the radius of the Earth at the equator, v_s is the velocity of the satellite which is the unknown, m_e is the mass of the Earth which is 5.392×10^{24} kg, G is the gravitational constant which is 6.67384×10^{-11} m^3 kg s^2, and r_s is the height of the satellite above the surface to be equal.

The idea is to find the velocity we require the satellite to be traveling at to achieve an orbit at r_s. To do this, we equate the two equations in (14.17) and rearrange, to obtain

$$v_s = \sqrt{\frac{Gm_e}{r_e + r_s}}. \tag{14.18}$$

Therefore, the significance of (14.18) is that the only factor that needs to be correct to achieve the orbit we desire is the velocity of the satellite.

We now consider three different types of orbits that will affect the quality or the spatial resolution of the data from the satellites. These three orbits are **sun-synchronous**, **geosynchronous** (also referred to as **geostationary**), and are often referred to as GEO, and **low inclination, low Earth orbit**, often referred to as LEO. As the next generation of observing networks is being designed at the time of writing in the Spring of 2022, there is much debate about how to optimize the combinations of GEO and LEO to meet the geophysical prediction needs.

There are three characteristics that we quickly introduce here about the orbits, to be able to differentiate between the three just mentioned:

- **Orbit Inclination:** This refers to the angle of the orbit relative to the equator.
- **Orbit Period:** This is how long it takes the satellite to complete a full orbit.
- **Orbit Node:** The ascending node is when the satellite travels from the South Pole to the North Pole, while the descending node is when the satellite travels from the North Pole to the South Pole.

Taking each of the three orbits mentioned above in turn, then the **sun-synchronous** orbits are a low Earth orbit, approximately 800 km, and have a high inclination, polar orbit; the orbit period is set based upon a fractional rotation of the Earth underneath the satellite. The satellite also passes over the equator at the same local time on its ascending node. The orbital plane for the satellite is consistent in position to the sun, which then enables the satellite to provide consistent global sampling daily.

Geosynchronous orbits are high Earth orbits which are approximately 35,000 km above the surface. The inclination for these satellites is zero as they stay above the equator, while the angular velocity of the satellite is equal to the angular velocity of the Earth, which implies that the relative velocity of the

satellite to the Earth's surface is zero. This implies that the orbit is one day. An advantage of this orbit is that it allows for good local temporal resolution below the satellite; however, multiple satellites are required for global coverage, but this orbit is weak for viewing poles because of the high viewing angle.

Finally we consider the **low-inclination** orbits that are generally circular orbits at which the inclination is set to study a specific question. However, these orbits are not sun-synchronous; as such, there is diurnal information in the observations. The relative position to the sun can differ, which results in complicated satellite maneuvers in space.

There are other types of orbits, where the GPS's satellites are in a circular medium Earth orbit at around 2000 km. If we consider the Deep Space Climate Observatory (DSCOVR), this space weather satellite is 1.6 million kilometers away from the Earth at the Lagrange 1 point.

We move on to instrument characteristics, where we have either active or passive instruments. **Active** instruments are those that measure the return reflected by an observed feature from an actively emitted, effectively man-made, source. Examples of these type of sensors are radar and lidar. A **passive** instrument or sensor measures the radiation emitted, or reflected, from the observed feature from a natural source, generally the Sun and Earth. Examples of these types of sensors are radiometers.

Fundamentally the sensors/instruments on the satellites are measuring radiances which in wavenumber space are given by mW m^2sr cm^{-1}, or energy per second (mW) per unit area of solid angle viewing, which are referred to as steradian per wavenumber cm^{-1}. We also have irradiance, which is the radiance integrated over the viewing area and wavenumber.

The last feature we consider of the sensor on the satellite is its **resolution**. There are four forms of resolution to consider: spatial, temporal, spectral, and radiometric. However, we have to realize that at the basic level of describing what a satellite sensor does, they are measuring the number of photons hitting a detector over a given time. The detectors have some quantifiable accuracy of how many photons it can detect over an integration period.

The spatial resolution of an instrument is determined by the observations that are made over a viewing cone, which is referred to as the Instantaneous Field of View (IFOV). The width of the cone affects the amount of photons that reach the detector. The temporal resolution is the amount of time between observations of the same area. This affects the observation accuracy because to achieve a higher temporal resolution, we require higher orbits. The spectral resolution is as a result of the satellites measuring over spectral response function. The spectral resolution is analogous to vertical resolution. Finally, radiometric resolution is the amount of pieces of data that can be discerned, which is to do with the bits of information that make up the observation.

Given the characteristics of the satellite, we move on to consider different types of radiation that the sensor observe but also which geophysical variables they are sensitive to.

14.2.3 Infrared

The infrared component of the electromagnetic spectrum is comprised of three types of waves: short waves that start at approximately 3811 K with wavelength 0.76 μm, medium wave that start at approximately 1450 K with wavelength 2 μm and finally the long waves that start at approximately 728 K and ranges between 4 and 10^{-3} μm. However, the infrared radiation coming from the sun means that it is the long wave infrared radiation that is coming from the Earth that is of interest from a remote sensing point of view.

How do we use infrared information? The theoretical emission of the Earth is at about 280 K, if the observations have warmer values than this, then we expect them to be of the heat of the sun directly,

or greenhouse effect. If there are areas where the emission is less than 280 K, then we have absorption occurring. The major absorbing constituents in the infrared are carbon dioxide, water vapor, ozone, and methane. When these constituent are absorbing the radiation, there is a drop as we said in the radiance, and as such these parts of the spectrum are not of much use to use; however, in the regions between these drops in emitted radiation, we can observe different atmospheric, ocean, and land variables. These gaps in the spectrum are referred to as **windows**. These **windows** are part of the electromagnetic spectrum that are almost clear of absorption by atmospheric gases and as such enable the atmosphere to appear almost transparent [211].

14.2.4 Microwave

Microwave passive measurements respond to many different geophysical variables such as surface albedo, cloud cover/layers, cloud optical thickness, cloud top height, cloud top pressure, net heat flux, ocean color/chlorophyll, vegetative index, cloud base height, ice surface temperature, land surface temperature, snow cover/depth, sea surface temperature, soil moisture, cloud liquid water, precipitation type and rate, precipitable water, sea surface winds, cloud ice water path, surface wind stress, atmospheric vertical moisture, temperature profiles, surface pressure, auroral energy deposition, auroral imagery, electron fields, energetic ions, geomagnetic field, long wave radiance, short wave radiance, ozone total column, ozone profile, aerosol optical thickness, particle size, and solar irradiance, to name some of them. The more commonly used passive microwave channels to observe different geophysical channels are summarized in Table 14.1.

It should be noted here that there are still microwave frequencies available above 183 GHz, but at the moment there has not been a satellite with a sensor at these frequencies that has been flown in space.

Another set of observations that are partially microwave based as those associated with the Gravity Recovery and Climate Experiment (GRACE) consists of two identical spacecraft that fly about 220 km apart in a polar orbit 500 km (310 miles) above Earth. GRACE maps Earth's gravity field by making accurate measurements of the distance between the two satellites, using GPS and a microwave ranging system. The results from this mission yield crucial information about the distribution and flow of mass within Earth and its surroundings.

Table 14.1 Different Microwave Frequencies Used to Remote Sense Geophysical Variables and Their Associated Observed Variables.

Frequency (GHz)	Geophysical Variables Observed
1.4	Soil moisture, ocean salinity
6 and 10	Heavy precipitation, soil moisture, sea surface temperature
19	precipitation, soil moisture, surface roughness (wind speed)
22	Total column water vapor
37	Precipitation, windspeed, surface
50–60	Temperature sounder
85	Precipitation
118	Temperature sounding
150	Precipitation, ice
183	Water vapor sounding

The gravity variations studied by GRACE include: changes due to surface and deep currents in the ocean; runoff and ground water storage on land masses; exchanges between ice sheets or glaciers and the ocean; and variations of mass within Earth. Another goal of the mission is to create a better profile of Earth's atmosphere. GRACE's terrestrial water storage estimates have been assimilated into the North American Land Data Assimilation System (NLDAS); see [228].

14.2.5 Visible

There are different types of observing mechanism for satellite remote sensing. The first of these is the visible imagery. The first imaging sensor measured radiation in the visible band of the electromagnetic spectrum. Imagery associated with visible radiation offers the highest spatial resolution and can provide views of the Earth that closely match what we perceive as visible. In visible imagery we can clearly see land, oceans, and clouds from which we can use this information. Until recently it was thought that visible channels were only of use during daylight hours, but recent work at the Cooperative Institute for Research in the atmosphere has shown that it is possible to obtain visible imagery of cloud through using **moonlight** [299,301,407]. This research is still in the early stages of development and there have not been any experiments to assimilate these type of observations yet.

14.2.6 Lidar

Lidar stands for light detection and ranging, although it is sometimes referred to as light imaging, detection, and ranging. Lidars use ultraviolet, visible, or near infrared light to image objects. From a geophysical point of view, lidar can target rain, chemical compounds aerosols, clouds, rocks. These types of sensor are active instruments as they emit a near ultraviolet/visible/near-infrared pulse and measure the backscatter, through either Rayleigh, Mie, Raman, or fluorescence scattering. For more details about these different forms of scattering, see [211].

Lidar has been used to create high resolution digital elevation maps from both airborne, where the lidar is attached to a plane during flight, and terrestrial lidar; which is where the lidar is stationary on the ground have greatly advanced the field of geomorphology. Lidar can also detect subtle topographical features. Lidar is also used in tectonophysics for detecting faults and for measuring uplift. Airborne lidar are able to monitor glaciers and are able to detect subtle amounts of growth or decline. ICESat contains the Geoscience Laser Altimeter System (GLAS), which is used to monitor the glaciers on Earth.

Scatterometer

A **radar scatterometer** is designed to determine the normalized radar cross section of the surface. Scatterometers work through transmitting a pulse of microwave energy toward pulse of microwave energy toward the Earth's surface and then measuring the reflected energy. The primary observation that comes from spaceborne scatterometry has been the measurement of near surface winds over the oceans. Scatterometers can also be used in the study of soil moisture, polar ice, and global environmental change.

14.2.7 Global Positioning System

The basis of this active instrument is that the characteristics of a GPS signal is emitted from another satellite is measured and used to determine geophysical information. One such example of this is through **radio occultation**, which is a vertical profile of refractivities that is related to temperature

and humidity. There is also GPS surface wind observations that are possible from this system of satellites where the surface roughness is determined by measuring GPS signals reflected off of the ocean surface.

The GPS consists of 29 satellites that are distributed in roughly six circular orbits planes at approximately 55° inclination at 20,200 km altitude and have an approximately 12-hour orbit. Each of the GPS satellites transmit signals at two at what are referred to as L-band frequencies of 1.57542 and 1.227 GHz. The satellites that make up the GPS network are referred to as Global Navigation Satellite Systems (GNSS), but there are also low Earth orbiting (LEO) satellites that play an important part in the radio occultation concept.

An occultation occurs when a GNSS satellite rises or sets across the limb with respect to a LEO satellite. A ray passing through the atmosphere is refracted due to the vertical gradient of refractivity which is a function of density and moisture. During an occultation, which lasts approximately 3 minutes, the ray path slices through the atmosphere. The raw measurement from the occultation is the change of the delay of the signal between the GNSS satellite and the LEO satellite, where the measurement includes the effect of the neutral atmosphere and the ionosphere.

Since the first edition the Constellation Observing System for Meteorology, Ionosphere, and Climate, which is referred to as COSMIC, 2, so COSMIC-2 has launched with six remote-sensing smallsats that form a network. This constellation of COSMIC satellites circles the equator at approximately 17,000 miles per hour. In recent years a private company called Spire has been launching smallsat radio occultation satellites to increase these observations. In [47] an assessment if GNSS radio occultations data from Spire is assessed in the United Kingdom's Met Office's NWP system and it is shown that assimilating the data from Spire gave a substantial benefit to forecasting. However, in [47] they state from another study that between 16,000 and 20,000 globally distributed occultations per day be the minimum requirement for the global observing system, in late 2020 it was only approximately 3,600. In [496] the use of radio occultation is shown as a basis for an observation of space weather.

The actual concepts of how to extract the different atmospheric variables from these types of observations are quite complicated and would take too long to explain here; however, for a full description of how the occultation works at extracting meteorological variables, please see [81].

14.3 Quality Control

Once the type of observations have been selected that are to be assimilated, we need to ensure that the observations are correct to within some parameters. The process of screening the observations is referred to as **quality control**. In this section we briefly present four mechanism for quality checking observations that are used in many research and operational data assimilation centers.

The first technique is **data thinning**. This is quite often applied to satellite observations, especially to hyperspectral satellites. Data thinning occurs because of the current need for the observational errors to be uncorrelated. To achieve this aim, the data are screened so that there are separated by a specific horizontal length and the observations in between are discarded. Deciding how far this distance should be is up to the users.

The **gross error check** is a simple check that says that given the difference between the observation y and the model equivalent of y, usually $h(x)$, and given an estimation of the observational background error; if the square of the difference between y and $h(x)$ normalized by the sum of the observational

and background error variances is greater than three, which is equivalent to three standard deviations of the Gaussian distribution, then the observations is seen as an outlier. We have plotted a figure of where the three standard deviations are on a Gaussian distribution in Fig. 14.1 to illustrate how far into the tails of the distribution the cut-off point is.

Buddy check

The next form of quality control that we consider is the **buddy check**. The original buddy check was developed through the United Kingdom Meteorological Office as a method to check observations with surrounding observations to see that the observation is consistent with its *surrounding buddies*. The starting point is to assume that a small fraction of the observations are corrupted, and hence are worthless, but we assume that the remaining observations have Gaussian errors. We start by denoting O as the event that the observation is between o and $o + do$, and T is the event that the true state lines between t and $t + dt$. The summary of the derivation of the different forms of buddy check quality control measures are from [193,265].

We start the derivation of the first form of buddy check, which is referred to as the **Gaussian check**, by assuming that we have prior knowledge about the PDF of the true values, either from forecasts and/or climatological knowledge, where we shall refer to this state as the **background**, and this knowledge gives us a background state b, which is the best **prior** estimate of the true state t. This background state can be thought of as an additional observation with a Gaussian error distribution with variance V_b. In

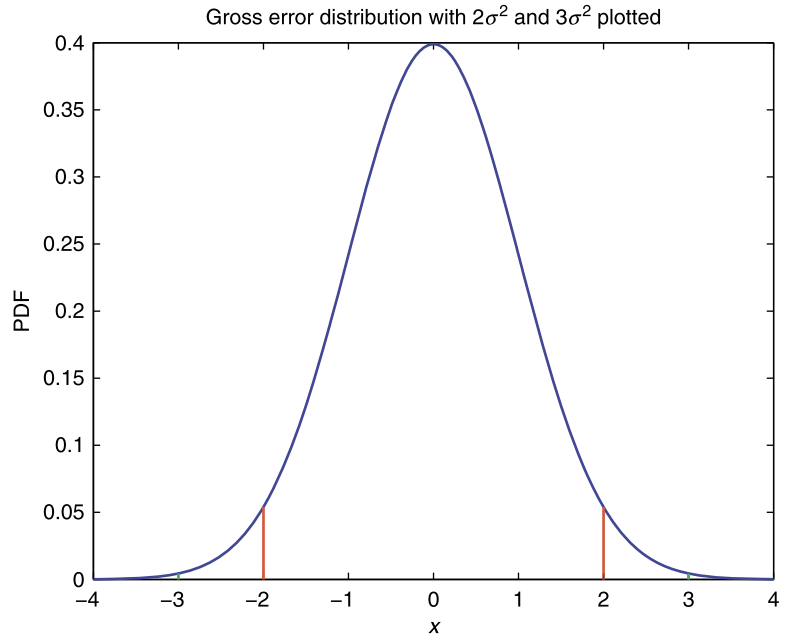


FIGURE 14.1

Plot of the $2\sigma^2$ (red) and $3\sigma^2$ (green) cutoffs for a standard Gaussian distribution.

[265] it is stated that the probability density of the true state is given by

$$P(T) = P_b(t)\,dt = N(t-b, V_b)\,dt, \tag{14.19}$$

where subscript b represents the background and $N(x, V)$ is a Gaussian distribution defined as

$$N(x, V) = \frac{1}{\sqrt{2\pi V}} e^{-\frac{x^2}{2V}}.$$

If we now assume that o is an imperfect observation of t, that has a Gaussian observational error distribution with variance V_o, then we have the conditional PDF of

$$P(O|T) = P_o(o)\,do = N(o-t, V_o). \tag{14.20}$$

If we now assume that we wish to consider all possible mutually exclusive events for T, then in the finite case of mutually exclusive events, we have that

$$P(O) = \sum_{i=1}^{N} P(O|T_i)\,P(T_i). \tag{14.21}$$

Now if we allow $N \to \infty$ continuously, then we have that

$$P(O) = \int_{t=-\infty}^{\infty} N(o-t, V_o)\,doN(t-b, V_b)\,dt. \tag{14.22}$$

Therefore, applying Bayes' theorem to events O and T, we obtain

$$P(T|O) = \frac{P(O|T)\,P(T)}{P(O)}. \tag{14.23}$$

Substituting (14.20) and (14.19) into (14.23), we obtain

$$P(T|O) = \frac{N(o-t, V_o)\,N(t-b, V_b)\,dt}{\int_{t=-\infty}^{\infty} N(o-t, V_o)\,N(t-b, V_b)\,dt}. \tag{14.24}$$

Due to the fact that the convolutions of two Gaussian distributions is also a Gaussian distribution, then we see that the denominator in (14.24) is the CDF of a Gaussian distribution evaluated between its limits, and we saw in Chapter 3 this is equal to one. Therefore, the remaining numerator is also a Gaussian distribution of the form

$$P(T|O) = N(a-t, V_a)\,dt = P_a(t)\,dt, \tag{14.25}$$

where a is referred to as the *analyzed value* in [265] and V_a is the analysis error variance, which are defined as

$$\frac{1}{V_a} = \frac{1}{V_o} + \frac{1}{V_b}, \tag{14.26a}$$

$$a = \left(\frac{o}{V_o} + \frac{b}{V_b}\right) V_a. \tag{14.26b}$$

In Fig. 14.2 we have plotted four different values for o to show how the analysis distribution is affected by these values.

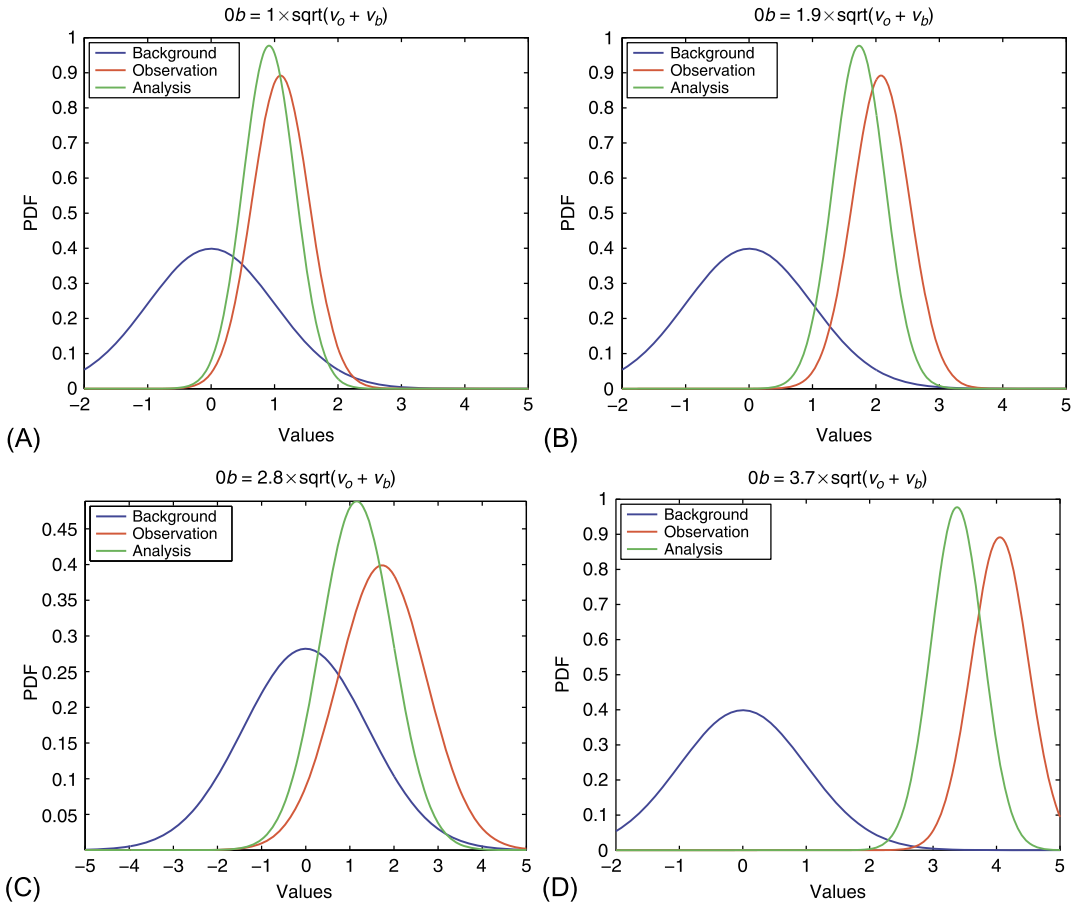


FIGURE 14.2

Plot of the effect of the different degrees of accuracy of the observations on the analysis distribution.

If we now let G represent the event that the observation has a gross error and $\overline{G}$ represent that the observation does *not* have a gross error, and we assume that plausible values with gross errors are equally likely with a constant probability of k which is independent of the true value t, and if we use the intersection operator $\cap$ to represent *and*, then we have that

$$P(O|T \cap G) = P(O|G) = kdo, \tag{14.27}$$

within the range, width of $1/k$, of plausible values, and by

$$P(O|T \cap G) = P(O|G) = 0, \tag{14.28}$$

outside that range.

By assuming that the prior (background) distribution has negligible small probabilities for implausible values, then we can neglect (14.28) and use (14.27) only for the gross error distribution. Given these assumptions the complete observational error distribution is

$$\begin{aligned} P\,(O|\,T) &= P\,(O|\,T\cap\overline{G})\,P\left(\overline{G}\right)+P\,(\,O|\,T\cap G)\,P\,(G)\,, \\ &= \left[N\,(o-t,V_0)\,P\left(\overline{G}\right)+kP\,(G)\right]do = P_o\,(O)\,do, \end{aligned} \tag{14.29}$$

where V_o represents the observation error variance. In other words the expression on the left-hand side of (14.29) is saying what is the probability of our observation lying between o and $o+do$ is true given that the true state lies between t and $t+dt$.

We now consider the situation the other way around, where we now seek the probability that the true state lies between t and $t+dt$, given that our observations line between o and $o+do$. From Bayes' theorem the conditional event just described has a PDF of the form

$$P\,(T|\,O) = \frac{P\,(T|\,O\cap\overline{G})\,P\left(\overline{G}\middle|\,O\right)+P\,(T|\,O\cap G)\,P\,(G|\,O)\,doN\,(b-t,V_b)\,dt}{P\,(O)}, \tag{14.30}$$

where we have assumed a Gaussian distribution for the background error model and where the normalizing factor in (14.30) is given by

$$P\,(O) = \left[N\,(o-b,V_o+V_b)\,P\left(\overline{G}\right)+kP\,(G)\right]do. \tag{14.31}$$

Fig. 14.3 shows a copy of the buddy check system from [265] for the same four values for the observation o as in Fig. 14.2, but now we have adjusted the PDF to allow for a constant distribution for the gross errors. We can see that as the observation becomes quite large compared to the background, this measure starts to give more credit to the background state and not the observation. This means that it is able to screen some bad observations compared to the background state. Given all of the probability theory derived above from [265], we are now able to introduce two forms of quality control checks; the first is referred to as the **comparison with background**. The motivation in [265] for this measure is to test the quality of the surface observations coming from ships. The authors start by introducing the notation of E^2 for the observation variance and F^2 for the background variances, and denote position, surface pressure, wind, and temperature by p, s, u, and T, respectively, and if we allow for two sources of gross errors G_p and G_s, then if we consider the surface pressure variable, we have that

$$P\,(O_s\,|\,T_s) = \left[N\left(o_s-t_s,E_s^2\right)P\left(\overline{G}_p\right)P\left(\overline{G}_s\right)+k_s\left(1-P\left(\overline{G}_p\right)P\left(\overline{G}_s\right)\right)\right]so. \tag{14.32}$$

We can write similar expression to that in (14.32) for the wind in terms of the absolute error vector error, which is defined as $|u_o-u_t|$, and temperature. If we now apply Bayes' theorem to G_p, then the result is

$$P\left(G_p\middle|\,O_s\cap O_u\cap O_T\right) = \frac{P\left(O_s\cap O_u\cap O_T\,|\,P_p\right)P\left(G_p\right)}{P\,(O_s\cap O_u\cap O_t)}. \tag{14.33}$$

If we now assume that the various instrumental and gross errors are independent, we have

$$\begin{aligned} P\,(O_s\cap O_u\cap O_T) &= P\left(O_s\,|\,G_p\right)P\left(O_u\,|\,G_p\right)P\left(O_T\,|\,G_p\right)P\left(G_p\right) \\ &+ P\left(O_s\,|\,\overline{G}_p\right)P\left(O_u\,|\,\overline{G}_p\right)P\left(O_T\,|\,\overline{G}_p\right)P\left(\overline{G}_p\right). \end{aligned} \tag{14.34}$$

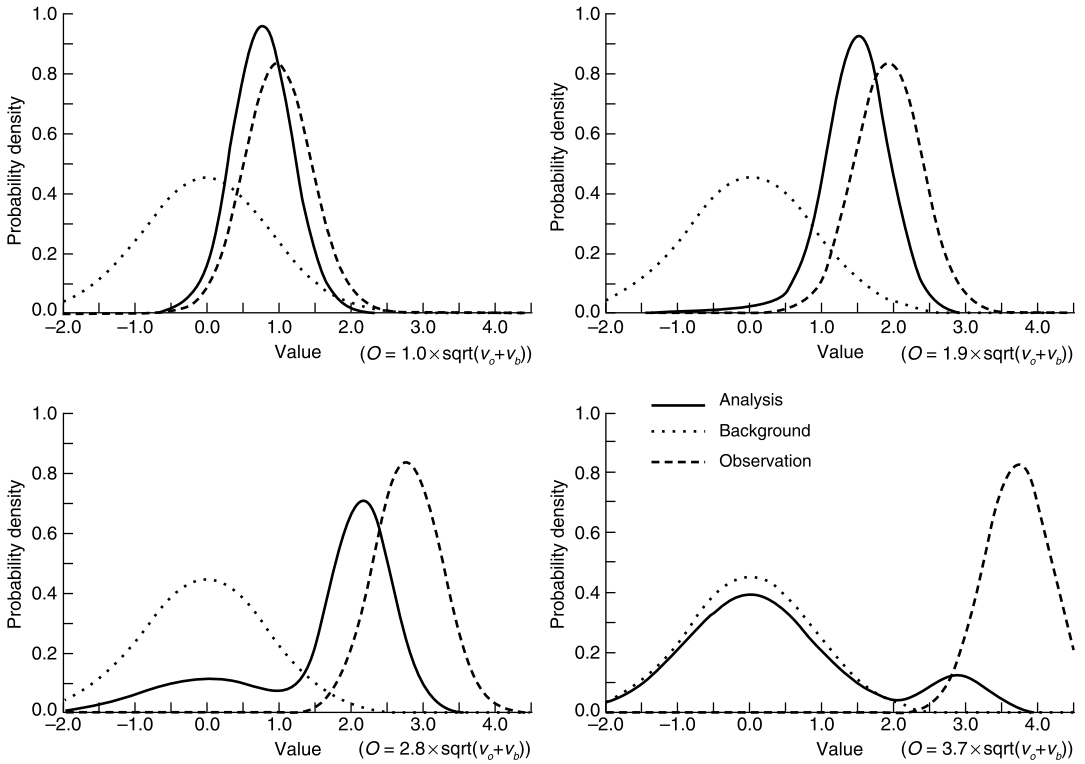


FIGURE 14.3

Copy of figure 2 from [265] illustrating the buddy check system.

It is possible to follow the same argument set out for G_p for G_s and obtain a similar expression to that in (14.33) but with G_s.

An important feature that we have to take into account is that while G_p and G_s are a priori independent events, it is not guaranteed that $G_p \mid O$ and $G_s \mid O$ are independent, therefore it is important to calculate these posterior probabilities that could be needed for future checking.

We need to carry out a final check, before we move on to the buddy check quality control measure, which is based on checking the different types of observations independently, without explicitly considering errors such as G_p that relate them. Thus we need to calculate the probability that surface pressure observation is unaffected by gross errors from *either* G_p or G_s, which results in

$$P\left(\overline{G}_p \cap \overline{G}_s \mid O_s \cap O_u \cap O_T\right) = \frac{P\left(O_s \cap O_u \cap O_T \mid \overline{G}_p \cap \overline{G}_s\right) P\left(\overline{G}_p \cap \overline{G}_s\right)}{P\left(O_s \cap O_u \cap O_T\right)}. \tag{14.35}$$

We now move on to the **buddy check** quality control measure, which is to check observations with surrounding observations to make sure that all of the observations are consistent with each other.

We shall start with the two observation formulation of the buddy check, which is what appears in [265], and then show the multivariate formulation for buddy check from [193]. Let O_1 and O_2 be two observation events. The buddy check is useful because if there were no gross errors, then the events would not be independent which imply that

$$P\left(O_1 \cap O_2 \,|\, \overline{G}_1 \cap \overline{G}_2\right) \neq P\left(O_1 | \overline{G}_1\right) P\left(O_2 | \overline{G}_2\right).$$

Thus the joint probability above can be written as

$$P\left(O_1 \cap O_2 \,|\, \overline{G}_1 \cap \overline{G}_2\right) = P\left(O_1 | \overline{G}_1 \cap O_2 \cap \overline{G}_2\right) P\left(O2 \,|\, \overline{G}_2\right). \tag{14.36}$$

As a result of excluding gross errors here, all of the distributions in (14.36) are Gaussian, then we can use what is referred to as **statistical (optimum/optimal) interpolation**. We shall introduce the theory for this form of interpolation in the next chapter, but as a result of this theory we can express the PDF of (14.36) as

$$P\left(O_1 | \overline{G}_1 \cap O_2 \cap \overline{G}_2\right) = N\left(o_1 - a_1, E_1^2 + A_1^2\right) do, \tag{14.37}$$

where

$$w = \left(E_2^2 + F_2^2\right)^{-1} F_1 \mu_{12} F_2, \tag{14.38a}$$

$$a = b_1 + w\left(o_2 - b_2\right), \tag{14.38b}$$

$$A_1^2 = F_1^2 - w F_1 \mu_{12} F_2, \tag{14.38c}$$

where μ_{12} is the covariance between b_1 and b_2.

Multiobservations buddy check

We now move onto the multivariate formulation of the background and the buddy check. Before we start we need some important properties of PDFs. We start by considering a vector of observations $\boldsymbol{y}$ which has an associated multivariate mean vector $\boldsymbol{\mu}$ and covariance matrix, $\boldsymbol{\sigma}$ and we consider splitting $\boldsymbol{y}$ to two smaller vectors $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$, which then have associated mean vectors and covariance matrices which enables us to express these quantities as

$$\boldsymbol{y} = \begin{pmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \end{pmatrix}, \quad \boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{y}_2 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}, \tag{14.39}$$

where $\boldsymbol{\Sigma}_{11}$ and $\boldsymbol{\Sigma}_{22}$ are the covariance matrices of $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ of order k and $n-k$, respectively. $\boldsymbol{\Sigma}_{12} = \boldsymbol{\Sigma}_{21}^T$ are of dimensions $k \times (n-k)$ and contain the covariances of $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$.

There are many different properties for the conditional and marginal PDF of the multivariate Gaussian that are presented in Chapter 4 that are used in [193]. The important property that we shall restate here is the conditional PDF of $P\left(\boldsymbol{y}_1 \,|\boldsymbol{y}_2\right)$, which is given by

$$p\left(\boldsymbol{y}_1 \,|\boldsymbol{y}_2\right) = \nu\left(\boldsymbol{y}_1 \,|\, \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right),$$

where $\boldsymbol{\mu}_c$ is the conditional mean given by

$$\boldsymbol{\mu}_c = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12} \boldsymbol{\Sigma}_{22}^{-1} \left(\boldsymbol{y}_2 - \boldsymbol{\mu}_2\right),$$

and $\boldsymbol{\Sigma}_c$ is the conditional covariance matrix given by

$$\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}.$$

So far we have not assumed any distribution type for this analysis. That stops here, where we now assume that the background and the observed values are assumed to have multivariate Gaussian distributions given by

$$P(\boldsymbol{y}) = p(\boldsymbol{y})\,d\boldsymbol{y} = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{B}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{y}_b)^t\,\mathbf{B}^{-1}(\boldsymbol{y}-\boldsymbol{y}_b)\right\}d\boldsymbol{y}, \tag{14.40a}$$

$$P(\boldsymbol{y}_o\,|\boldsymbol{y}) = p(\boldsymbol{y}_o\,|\boldsymbol{y})\,d\boldsymbol{y}_o = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{O}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{y}_0)^t\,\mathbf{O}^{-1}(\boldsymbol{y}-\boldsymbol{y}_o)\right\}d\boldsymbol{y}_o, \tag{14.40b}$$

where $\mathbf{B}$ and $\mathbf{O}$ are the background and observational error covariance matrices, respectively, $\boldsymbol{y}$, $\boldsymbol{y}_b$, and $\boldsymbol{y}_o$ are the vectors of the true, background, and observed values, respectively, and n is the number of observations.

Therefore, the joint probability of the true state and the observed values, which is denoted by $P(\boldsymbol{y}\cap\boldsymbol{y}_o)$, is given by the multivariate version of Bayes' theorem as

$$\begin{aligned} p(\boldsymbol{y}\cap\boldsymbol{y}_o) &= p(\boldsymbol{y}_o\,|\boldsymbol{y})\,p(\boldsymbol{y}) \\ &= \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{B}+\mathbf{O}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\boldsymbol{y}_o-\boldsymbol{y}_b)^T(\mathbf{B}+\mathbf{O})^{-1}(\boldsymbol{y}_o-\boldsymbol{y}_b)\right\} \\ &\quad\times\frac{1}{(2\pi)^{\frac{n}{2}}\left|\left(\mathbf{B}^{-1}+\mathbf{O}^{-1}\right)^{-1}\right|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\boldsymbol{y}_o-\boldsymbol{y}_m)^T\left(\mathbf{B}^{-1}+\mathbf{O}^{-1}\right)^{-1}(\boldsymbol{y}_o-\boldsymbol{y}_m)\right\}, \end{aligned} \tag{14.41}$$

where

$$\boldsymbol{y}_m = \boldsymbol{y}_b + \mathbf{B}(\mathbf{O}+\mathbf{B})^{-1}(\boldsymbol{y}_o-\boldsymbol{y}_b). \tag{14.42}$$

The probability of $\boldsymbol{y}_0$ occurring is

$$P(\boldsymbol{y})\int p(\boldsymbol{y}\cap\boldsymbol{y}_o)\,d\boldsymbol{y} = \frac{1}{(2\pi)^{\frac{n}{2}}|\mathbf{B}+\mathbf{O}|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2}(\boldsymbol{y}_o-\boldsymbol{y}_b)^T(\mathbf{B}+\mathbf{O})^{-1}(\boldsymbol{y}_o-\boldsymbol{y}_b)\right\}. \tag{14.43}$$

There have been some manipulations of the properties of determinants and the reader is referred to Tarantola's 2005 book *Inverse Problem Theory and Methods for Model Parameter Estimation* [429] for the exact derivation.

Finally we can evaluate the conditional PDF of $\boldsymbol{y}$ given $\boldsymbol{y}_o$ through (14.41) and (14.43) as

$$p(\boldsymbol{y}\,|\boldsymbol{y}_o) = \frac{p(\boldsymbol{y}\cap\boldsymbol{y}_o)}{p(\boldsymbol{y}_o)} = \frac{1}{(2\pi)^{\frac{n}{2}}\left|\left(\mathbf{B}^{-1}+\mathbf{O}^{-1}\right)^{-1}\right|^{\frac{1}{2}}}\times\exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{y}_m)^T\left(\mathbf{B}^{-1}+\mathbf{O}^{-1}\right)(\boldsymbol{y}-\boldsymbol{y}_m)\right\}. \tag{14.44}$$

The expression in (14.44) is that of the multivariate background quality control check.

We now move on to the case where there are gross errors in the observations; effectively this is the summary of the derivation to the equations for multivariate version of the buddy check quality control measure.

Multi-observation buddy check quality control

As in the univariate part of this section, we shall assume that the "good observations" follow a Gaussian distribution, whereas the observations with gross errors follow a uniform distribution, which then implies that all plausible values are equally likely. We shall assume that the observational errors are independent. This then gives us

$$\begin{aligned} p\left(\mathbf{y}_o \mid \mathbf{y}\right) &= \prod_{i=1}^{n} p\left(y_{oi} \mid y_i\right), \\ &= \prod_{i=1}^{n} \left\{ p\left(y_{o,i} \left| y_i \cap \bar{G}_i \right.\right) P\left(\bar{G}_i\right) + p\left(y_{oi} \left| y_i \cap G_i \right.\right) P\left(G_i\right) \right\}, \\ &= \prod_{i=1}^{n} \left\{ p\left(y_{oi} \left| y_i \cap \bar{G}_i \right.\right) P\left(\bar{G}_i\right) + \kappa_i P\left(G_i\right) \right\}, \end{aligned} \tag{14.45}$$

where i is an index over the observations, G_i is the event that observation i contains a gross error, and $\bar{G}_i$ is the converse. For the analysis that follows, it is assumed that the PDF for an observation with gross errors, $p\left(y_{oi} \left| y_i \cap G_i \right.\right)$, is a constant and is equal to κ_i within an interval of width κ_i^{-1} and zero outside.

If we now consider the total density of the observations $\mathbf{y}_o$, we have the PDF defined as

$$\begin{aligned} p\left(\mathbf{y}_o\right) &= \int p\left(\mathbf{y} \cap \mathbf{y}_o\right) d\mathbf{y} = \int p\left(\mathbf{y}\right) p\left(\mathbf{y}_o \left| \mathbf{y} \right.\right) d\mathbf{y}, \\ &= \int p\left(\mathbf{y}\right) \prod_{i=1}^{n} \left\{ p\left(y_{o,i} \left| y_i \cap \bar{G}_i \right.\right) P\left(\bar{G}_i\right) + \kappa_i P\left(G_i\right) \right\} d\mathbf{y}. \end{aligned} \tag{14.46}$$

If we now expand the product in (14.46), then we have $2^n - 1$ different sets of possible results given n observations. The combinations are

$$\begin{aligned} S_0 &= G_n \cap G_{n-1} \cap \cdots \cap G_2 \cap G_1, \\ S_1 &= G_n \cap G_{n-1} \cap \cdots \cap G_2 \cap \bar{G}_1, \\ S_2 &= G_n \cap G_{n-1} \cap \cdots \cap \bar{G}_2 \cap G_1, \\ &\vdots \\ S_{2^n-1} &= \bar{G}_n \cap \bar{G}_{n-1} \cap \cdots \cap \bar{G}_2 \cap \bar{G}_1. \end{aligned} \tag{14.47}$$

For each S_i, i is written in binary and the bits numbered from 1 to n starting from the right. Bit j equal to 0 corresponds to G_j while 1 refers to $\bar{G}_j$. Therefore, S_0 refers to all the observations having a gross error while S_{2^n-1} refers to all the observations having no gross error.

We can therefore write (14.46) as a summation in terms of S_i due to the events now being unions rather than intersections, making (14.46)

$$p\left(\mathbf{y}_o\right) = \int p\left(\mathbf{y}\right) \sum_{i=1}^{2^n-1} p\left(p\left(\mathbf{y}_o \left| \mathbf{y} \cap S_i \right.\right) P\left(S_i\right)\right) d\mathbf{y}. \tag{14.48}$$

As we can clearly see in (14.48) that the summation in the integral is independent of the integrating variable, we can therefore remove it, leaving the integral of a PDF that is equal to 1. Therefore (14.48) can be written as

$$p\left(\mathbf{y}_o\right) = \sum_{i=1}^{2^n-1} p\left(p\left(\mathbf{y}_o \left| \mathbf{y} \cap S_i\right.\right) P\left(S_i\right)\right). \tag{14.49}$$

If we now assume that there are m good observations and that the remaining observations have gross errors, that are independent of each other, then after some introduction of notation in [193], the equation for the multivariate buddy check quality control measure is

$$\begin{aligned} p\left(\mathbf{y}_o \left| S_j\right.\right) P\left(S_j\right) &= P\left(S_j\right) \prod_{j=m+1}^{n} \kappa_j \int p\left(\mathbf{y}_1\right) p\left(\mathbf{y}_{o1} \left| \mathbf{y}_1\right.\right) d\mathbf{y}_1, \\ &= P\left(S_j\right) \prod_{j=m+1}^{n} \kappa_j \frac{1}{(2\pi)^{\frac{k}{2}} \left|\mathbf{B}_{11} + \mathbf{O}_{11}\right|^{\frac{1}{2}}} \times \exp\left\{-\frac{1}{2}\left(\mathbf{y}_{b1} - \mathbf{y}_{o1}\right)^T \left(\mathbf{B}_{11} + \mathbf{O}_{11}\right)^{-1} \left(\mathbf{y}_{b1} - \mathbf{y}_{o1}\right)\right\}, \end{aligned} \tag{14.50}$$

where

$$P\left(S_j\right) = \prod_{i=1}^{k} P\left(\bar{G}_i\right) \prod_{m=k+1}^{n} P\left(G_m\right). \tag{14.51}$$

Any combination S_j can reordered into the form used above; in general $\mathbf{y}_1$ is formed by selecting those observations assumed useful, and $\mathbf{B}_{11}$ and $\mathbf{O}_{11}$ are formed from the corresponding elements of $\mathbf{B}$ and $\mathbf{O}$.

The probability of combination S_j is

$$P\left(S_j \left| \mathbf{y}_o\right.\right) = \frac{p\left(\mathbf{y}_o \left| S_j\right.\right) P\left(S_j\right)}{p\left(\mathbf{y}_o\right)}. \tag{14.52}$$

It is possible to use these probabilities to evaluate the analysis distribution $p\left(\mathbf{y} \left| \mathbf{y}_o \cap S_j\right.\right)$ as

$$p\left(\mathbf{y} \left| \mathbf{y}_o\right.\right) = \sum_{j=0}^{2^n-1} p\left(\mathbf{y} \left| \mathbf{y}_o \cap S_j\right.\right) P\left(S_j \left| \mathbf{y}_o\right.\right). \tag{14.53}$$

It is also possible to calculate the probability of gross error in individual observations $P\left(G_i \left| \mathbf{y}_o\right.\right)$ through

$$P\left(G_i \left| \mathbf{y}_o\right.\right) = \frac{p\left(\mathbf{y}_o \left| G_i\right.\right) P\left(G_i\right)}{p\left(\mathbf{y}_o\right)} = \frac{p\left(\mathbf{y}_{o-i}\right) \kappa_i P\left(G_i\right)}{p\left(\mathbf{y}_o\right)}, \tag{14.54}$$

where $\mathbf{y}_{o-i}$ is the vector observations excluding i, and can be written as

$$P\left(G_i \left| \mathbf{y}_o\right.\right) = \frac{\sum_{j=0}^{2^n-1} \gamma_i\left(j\right) p\left(\mathbf{y}_o \left| S_j\right.\right) P\left(S_j\right)}{\sum_{j=0}^{2^n-1} p\left(\mathbf{y}_o \left| S_j\right.\right) P\left(S_j\right)}, \tag{14.55}$$

where

$$\gamma_i = \begin{cases} 1 & \text{if } G_i \in S_j, \\ 0 & \text{otherwise.} \end{cases}$$

The observation values in vector y_o are being compared with other to determine how likely each is to contain a gross error, a *buddy check*. The reader is referred to [193] for the finer details about implementing this approach.

14.3.1 Variational Quality Control

The variational-based quality control, as its name suggests, is a quality control technique that is associated with the variational data assimilation system; we shall introduce the variational data assimilation theory in Chapter 16. Variational quality control was introduced in [10] where the starting point in the derivation of the measure is again to assume a multivariate Gaussian distribution for the observational errors. In variational data assimilation we consider a cost function that is comprised of two components, one associated with the background errors and one with observational errors. This latter term is denoted by J_o and is defined as

$$J_o = \frac{1}{2} \left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}\right) \right)^T \mathbf{O}^{-1} \left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}\right) \right), \tag{14.56}$$

where $\boldsymbol{h}$ is referred to as the **observation operator**. This maps the model states to the observation space and location, in time and space; it can be nonlinear, as is the case with the satellite data, or linear with respect to say the radiosondes. If we assume that the observations are uncorrelated, then the matrix-vector multiplication in (14.56) collapses to

$$J_o = \sum_{i=1}^{N_o} \frac{1}{2} \left(\frac{y_i - \boldsymbol{h}_i\left(\boldsymbol{x}\right)}{\sigma_o} \right)^2, \tag{14.57}$$

where σ_o is the observational error standard deviation.

The basis of VAR-QC, as it is often called is again to follow the ideas from [193], where there are good observations that have a Gaussian error distribution associated with them and then there are observations that have a gross error and as such follow a different distribution. In the original formulation of VAR-QC [10] initially used a **flat distribution** for the gross error distribution. Therefore, the formulation of the probability of a good observation following the VAR-QC approach is

$$p^{QC} = (1 - A)\, N + A F, \tag{14.58}$$

where N and F represent Gaussian and flat distributions that are defined as

$$N = \frac{1}{\sigma_o \sqrt{2\pi}} \exp\left\{ -\frac{1}{2} \left(\frac{y - \hat{y}}{\sigma_o} \right)^2 \right\}, \tag{14.59a}$$

$$F = \frac{1}{D} = \frac{1}{2 d \sigma_o}, \text{ if } \left| y - \hat{y} \right| < \frac{D}{2}, \text{ else } 0, \tag{14.59b}$$

where y is the observed value, and $\hat{y}$ is the model equivalent. The flat distribution is defined over the interval D, that is centered at zero, which is a multiple d of the assumed observational error standard deviation.

To obtain the observational cost function, we take the negative logarithm of (14.56), which results in

$$J_o = -\ln p + c. \tag{14.60}$$

Therefore, by inserting (14.59a) into (14.60) and setting $c = -\ln\left(-\sigma_o\sqrt{2\pi}\right)$, we have

$$J_o^N = \frac{1}{2}\left(\frac{y-\hat{y}}{\sigma_o}\right)^2. \tag{14.61}$$

The gradient with respect to the modeled observed quantity, $\hat{y}$, is given by

$$\nabla_{\hat{y}} J_o^N = -\frac{1}{\sigma_o}\left(\frac{y-\hat{y}}{\sigma_o}\right). \tag{14.62}$$

If we now include the flat distribution part of the measure, then we obtain the cost function and its gradient for this formulation as

$$J_o^{QC} = -\ln\left(\frac{\gamma + \exp\left\{-J_o^N\right\}}{\gamma+1}\right), \tag{14.63a}$$

$$\nabla_{\hat{y}} J_o^{QC} = \nabla_{\hat{y}} J_o^N \left(1 - \frac{\gamma}{\gamma + \exp\left\{-Jo^N\right\}}\right), \tag{14.63b}$$

for $\left|y-\hat{y}\right| < \frac{D}{2}$ else $\nabla\hat{y} Jo^{QC} = \nabla\hat{y} Jo^N$, where γ is defined as

$$\gamma = \frac{A\sqrt{2\pi}}{(1-A)\,2d}. \tag{14.64}$$

Following (14.63b) it is possible to define the VAR-QC weight W^{QC}:

$$\nabla_{\hat{y}} J_o^{QC} = \nabla_{\hat{y}} J_o^N W^{QC}, \tag{14.65a}$$

$$W^{QC} = 1 - \frac{\gamma}{\gamma + \exp\left\{J_o^N\right\}} = 1 - P. \tag{14.65b}$$

Therefore, (14.65b) indicates that when the observations are likely to be incorrect then $P \approx 1$, while if the observations are quite accurate, then $P \approx 0$.

In [10] the authors mention that there are other models for the gross error that could be used; in particular they mention a model by Huber. Recently at ECMWF what is referred to as the **Huber Norm** [187] has been implemented into their observation quality control to allow better probabilities to be assigned to near outliers according to a Gaussian or a Gaussian plus a flat distribution control.

The definition for the Huber norm for a quality control measure is

$$P(y \mid x) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_o} \exp\left\{\frac{a}{2} - |a\delta|\right\} & \text{if } a < \delta \\ \frac{1}{\sqrt{2\pi}\sigma_o} \exp\left\{-\frac{\delta^2}{2}\right\} & a \le \delta \le b \\ \frac{1}{\sqrt{2\pi}\sigma_o} \exp\left\{\frac{b}{2} - |b\delta|\right\} & \text{if } \delta > b \end{cases}, \tag{14.66}$$

where $\delta = \frac{y-h(x)}{\sigma_o}$. For a description of the operational implementation of the Huber norm, see [432].

To finish this section we have plotted the cost functions for three different measures we have considered for the VARQC: the Gaussian only, the Gaussian plus a flat distribution, and the Huber norm to show how the spread of the three measures differ for the normalized differences in Fig. 14.4.

For a detailed description on the implementation and the effectiveness of the Huber norm as a quality control measure with the ECMWF operational numerical weather prediction system, the reader is referred to [432].

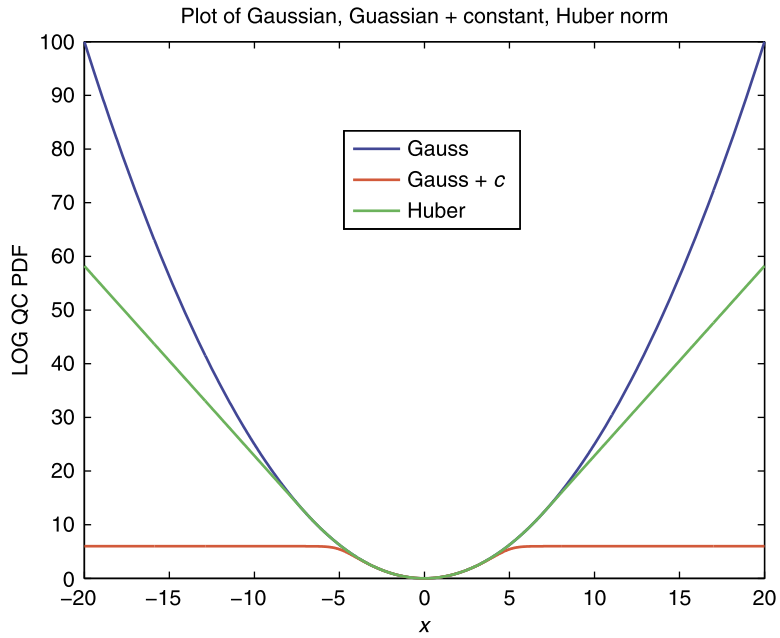


FIGURE 14.4

Plot of three possible quality control measures: Gauss (blue), Gauss + constant (red), and the Huber norm (green).

However, gross errors may not be the only problem with different types of observations: there could be a **bias**, which is not a random effect. In the next section we present methods to correct the observations for these biases.

14.3.2 Variational Bias Correction

When dealing with satellite observations, we have to be aware of biases that can be introduced into the innovation, which is the difference between the observed value and the model equivalent. Source of biases can vary: for example, forms of instrument errors, where this could be a scan bias, or orbital bias. As the innovation is a function of the modeled equivalent to the radiance, brightness temperature, then biases could be introduced in the radiative transfer model. Another source of biases is through what is referred to as **cloud contamination**, where presently many operational numerical weather prediction centers only assimilate cloud free data. However, at the time of writing, there is a lot of research into being able to assimilate cloudy radiances. If, however, a cloud has made it through a screening process, this can cause a bias in the radiance. Finally another source of bias is if the radiance is resolving a dynamical scale that the radiative transfer, as well as the numerical model, is not resolving, sometimes referred to as a subgrid scale.

Bias can be seen as a systematic, which means **non-random**, difference between the time series of observed and computed brightness temperatures at a given spatial location. These systematic errors can be a function of space, time, orbit, scan, radiative transfer parameters, and so on. Given the possibilities just described this then the question is: can we ignore biases? The answer is **no**, because data assimilation systems are designed to correct small random errors, not large systematic errors. Another important point is that radiances now account for more than half of the global forecast skill in operational numerical weather prediction; we shall explain forecast skill in Chapter 17.

Therefore, how do we detect radiance bias? One approach is to check to see that if the analysis from the data assimilation scheme is optimal then the associated innovations are **white**, which refers to the innovations be Gaussian distributed, in time, which implies that the innovations are not serially correlated, and they should have a mean innovation close to **zero**. If it is the case that the residual has a non-zero mean, which could be a spatial pattern that persists in time, then this indicates that the innovations are **biased**.

Some of the reasons why we wish to separate the biases are as follows:

- **(i)** We wish to understand the origins of the bias and then in theory correct the instrument/radiative transfer/numerical model at the source.
- **(ii)** We do not wish to apply a correction to **unbiased** satellite data if is the numerical model that is biased. If we were to do this, then:
  - **(a)** we would be reinforcing the model bias and would be degrading the analysis fit to **other** observations.
  - **(b)** We would produce a **biased analysis**, which would be bad for reanalysis and climate applications.

Variational bias correction, again as the names suggested with the quality control technique, is based upon minimizing a cost function. The idea for variational bias correction came about through [89,94]. We start by defining a modified observation operator that includes bias correction parameters which are linear regression coefficients; again we shall go into detail about linear regression in the next chapter,

of the form

$$\widehat{\boldsymbol{h}}(\boldsymbol{x},\boldsymbol{\beta}) = \boldsymbol{h}(\boldsymbol{x}) + \sum_{i=0}^{N} \beta_i \, p_i(\boldsymbol{x}), \tag{14.67}$$

where β_i are the regression coefficients and $p_i(\boldsymbol{x})$ is a third-order polynomial for scan predictors and are global polynomials.

Given the expression for the biased observation operator in (14.67) we now solve the following cost function, which is referred to as being **bias aware**:

$$J(\boldsymbol{\beta}) = \frac{1}{2}\left(\boldsymbol{y} - \widehat{\boldsymbol{h}}(\boldsymbol{x},\boldsymbol{\beta})\right)^T \mathbf{R}^{-1}\left(\boldsymbol{y} - \widehat{\boldsymbol{h}}(\boldsymbol{x},\boldsymbol{\beta})\right) + \frac{1}{2}\left(\boldsymbol{\beta} - \boldsymbol{\beta}_b\right)^T \boldsymbol{\Gamma}^{-1}\left(\boldsymbol{\beta} - \boldsymbol{\beta}_b\right) + \frac{1}{2}\left(\boldsymbol{x} - \boldsymbol{x}_b\right)^T \mathbf{B}^{-1}\left(\boldsymbol{x} - \boldsymbol{x}_b\right), \tag{14.68}$$

where the subscripts b represent the *background information* that we already have for the bias correction coefficients β_i and the model state, $\boldsymbol{\Gamma}$ is the covariance matrix between the regression coefficients.

For a good example of the use of variational bias correction, see [91]. The variational bias correction has also been applied to aircraft temperature biased data [501].

14.4 Summary

In this chapter we have introduced some of the types of observations of geophysical variables that may be of use to different data assimilation systems. We have introduced the concept of direct and indirect observations; for the latter we introduce remote sensing, where we also gave a brief introduction to radiative transfer theory. We have also introduced the concepts of passive and active sensors. We have presented different mechanism for quality controlling observations from an operational center's requirements. For initial research requirement, there are many other forms of quality control that can be applied. For the quality control of ocean sea surface temperatures there is a program at NOAA called iQuam, which stands for the in situ STT Quality Monitor [485], where they apply a Bayesian reference and the buddy check system. In [413] there is a summary of different types of quality control measures that are available for different applications.

We have also introduced the Huber norm as a method of changing the shape of the posterior PDF in a variational quality control that then enables more observations that have been assigned smaller probabilities with respect to this measure, but that would have been discarded with the just Gaussian or the Gaussian plus a constant for the posterior distribution. The Huber norm-based variational quality control measure is a method of allowing a light weak form of non-Gaussianity into the data assimilation system.

Another approach that we did not introduce in this chapter, but give reference to here is that of the adaptive buddy check. This technique is presented in [90], where it is shown that assimilating extra observations that would be rejected by the full Gaussian case had a positive impact on a rerun of a blown forecast of the big winter storm on December 27, 1999.

A final thing to note here is that when we are introducing new observations in to a data assimilation scheme, we need to have developed the observational (forward) operator, its tangent linear model, and its adjoint. All of these models are essential for data assimilation. It is possible to approximate

the observational operator's Jacobian by taking the difference between two perturbed versions of the operator. We shall see this applied with the maximum likelihood ensemble filter (MLEF) in Chapter 20.

Given the numerical model and now the selection of observations, the next step is to decide which data assimilation algorithm to apply to the problem that you are considering. We start to introduce data assimilation algorithms in the next chapter, where these approaches are referred to as empirical along with the statistical interpolation approaches.

This page intentionally left blank

CHAPTER

Non-Variational Sequential Data Assimilation Methods

15

Contents

So far we have introduced many techniques for numerically modeling different geophysical phenomenon, along with developing probability, control, optimal control, and calculus of variation, theory, that are the basis of data assimilation. In the previous chapter we introduced different types and sets of observations, be they direct or indirect, that are available for different geophysical situations. However, keeping the numerical model and the observations separate does not help us to provide better forecasts of the geophysical system of interest. We need to develop methods to combine the two, using the strengths of both, while trying to minimize the weaknesses of both at the same time. This is **data assimilation**.

In this chapter we shall introduce the original, and simpler, forms of data assimilation, that are still important tools in combining the models and the observations, and in use today for certain types of problems. There are a lot of different complexities involved with the variational and ensemble techniques for data assimilation, which we shall go into more detail about in their respective chapters, and sometimes for nonlinear systems it is not efficient to use the more complex systems. Another advantage of the simpler methods is if a data assimilation system is being developed from scratch then it can take a

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00019-X

long time; however, if a forecast is required before the full system is developed, then the techniques developed in this chapter can give indications of the sensitivity of the numerical model to the introduction of the observational information.

15.1 Direct Insertion

As the name suggests, the direct insertion (DI) technique is where a direct observation of the model field is used to replace the numerical model's value at that point. The observations can be indirect observations originally, but must be transformed into the model variable before insertion. The fundamental assumption made here is that the observation is **perfect**. That is to say that the observation is the value that the *true state* has at that time and place. Note that DI can also apply to observations that are not at model times or grid points but that have been interpolated to the point in space and or time.

The advantage of the DI technique is that it is easy to implement: you simply replace the current numerical model's value in an array with the observed value. However, with DI it is quite often the case that the observation is a point observation and as such does not have much influence over a larger area.

An example of DI in an oceanography content is presented in [364], using the Environmental Fluid Dynamics Code (EFDC) model that simulates the dynamic process of Galway Bay in Ireland. The EFDC solves the three-dimensional, vertically hydrostatic, free surface, turbulent averaged equations of motions for a variable density fluid. The module uses a sigma vertical coordinate and curvilinear, orthogonal horizontal coordinates. The model has been applied to a variety of modeling studies. The EFDC could provide the capability of internally linking four major modules: hydrodynamics, water quality, sediment transport, and toxins.

In the DI study in [364], the authors use a simulation domain that has 380,241 grid points, which equates to a resolution of 150 meters. Their physical domain is from (−9.71891E, 52.97371N) (left at the bottom) to (−8.87716E, 53.03773N) (right on the top), which effectively corresponds to Galway Bay. However, for the DI experiments they use the smaller domain of a 4.65 km × 4.65 km square area.

In [364] the authors perform a study to investigate the sensitivity of the model at the surface layer to the pseudo-observations that were set to be a constant at all locations and at all times. The aim of the study was to test the robustness of the model to a more advanced data assimilation system, as we mentioned at the beginning of this section. The authors tested different time intervals between the insertion of the surface velocity observations and saw that the model did respond to the insertion of what they refer to as extreme constant values for the winds.

The conclusions from [364] were as follows:

(1) The EFDC model is robust to frequently combine extreme constant measurement states with the model background states. This is of great importance when we use in situ measurement data to update our background model states, since there are always noises in the real measured data set.

(2) The model is sensitive to the data assimilation interval: the shorter the DI data assimilation interval, the stronger the influence on the model background states.

(3) The DI data assimilation has an impact on the surface area outside the data assimilation domain. The main difference comes out in the data assimilation domain. The degree of influence of DI data assimilation decreases with the distance to data assimilation domain.

The last two points of the conclusions are important, as they indicate that DI can have an influence through the model to other points without a decorrelation length scale being applied.

An example of DI being used in a hydrometeorological case can be found in [134]. In this study the numerical model was a snow evolution model called SNOWMODEL, developed at Colorado State University by Dr. Glen Liston—see references in [134] for more details about SNOWMODEL. In [134] the authors were concerned with assimilating snow cover data from MODIS at a high resolution, 500 m, with snow water equivalent (SWE) from AMSR-E at 25 km resolution. The problem here is that the snow cover observations could be considered as a binary, discontinuous, indirect observations; by this it is meant that the presence of snow in the MODIS observations informs us that the SWE > 0, or if there is no snow cover, then the SWE should be equal to zero. The observation is referred to as a binary observation due to it taking the value 1 for snow presence and 0 if there is not; however, it should be noted that the actual values of the MODIS snow cover observations take a different value to these, but from a data assimilation prospective they are equivalent.

The DI technique used in [134] was actually not quite a DI technique, in that the SWE observations covered multiple grid cells and as such it was a combination of these cells that were altered to match the AMSR-E observations. However, the MODIS snow cover information informed the iterative scheme which model cells inside the larger AMSR-E observation had snow cover, and as such are available to be adjusted to match the total SWE from the observation. This is considered a DI scheme in that the model is forced to match the snow cover observations from MODIS, as well as the SWE over the AMSE-R observation coverage area.

It was shown that using a combination of the two observations in the iterative scheme enabled SNOWMODEL to better match the 24-hour snow cover observations, where there was no cloud interference, than when the model was run on its own. However, as the reallocation or removal of the SWE inside of each ASMR-E area was arbitrary and evenly divided up among the remaining snow-covered cells, that is when the performance compared to snow depth data from snow observing sites in the area was not as good as an agreement, suggesting that snow depth observations need to be assimilated as well.

Another example of a DI application can be found in [418] where the authors assimilated images of sediment from the Sea-Viewing Wide Field Spectrometer over Lake Michigan. Images similar to the MODIS snow cover are not differentiable observation of the state variables, but they hold important information about the geophysical system that could be lost if there is not a way to introduce that information into the models.

Another cryosphere application of DI can be found in [486], where the authors compare the DI technique with the ensemble Kalman filter to assimilate MODIS snow albedo and snow covered fraction in a region of China during the winter of 2008–2009.

A non-Earth-bound example of using DI for a radiation belt particle distribution model can be found in [313], where the authors compare DI with an extended Kalman filter to insertion of simulated flux measurements.

There are many more examples of DI data assimilation techniques being used that we could summarize, but we now move on to another form of an insertion method: **nudging**.

15.2 Nudging

The application of **nudging** to a numerical model is seen as an empirical analysis scheme [208]. The method of nudging is also sometime referred to as Newtonian relaxation [178].

Hoke and Anthes [178] consider the equations of motion for an incompressible fluid with a free upper surface of height h on an f-plane, where it is assumed that no variation in the y direction are permitted, and that the system is linearized with respect to a motionless height, H. Given these assumption, and with the inclusion of a nudging term that forces the model to tend toward the *perfect* observations, the equations used to test this relaxation/nudging technique are:

$$\frac{\partial u}{\partial t} - fv + g\frac{\partial h}{\partial x} - G_u\left(u_{obs} - u\right) = 0, \tag{15.1a}$$

$$\frac{\partial v}{\partial t} + fu - G_v\left(v_{obs} - v\right) = 0, \tag{15.1b}$$

$$\frac{\partial h}{\partial t} + H\frac{\partial u}{\partial x} - G_h\left(h_{obs} - h\right) = 0, \tag{15.1c}$$

where (G_u, G_v, G_h) are the **nudging coefficients** and $(u_{obs}, v_{obs}, h_{obs})$ represent the **perfect** observations of the u, v, and h fields, respectively. If the boundaries are periodic, then it is possible to express the dependent variables in terms of Fourier series can be expressed as

$$\begin{pmatrix} u \\ v \\ h \end{pmatrix} = \sum_m \begin{pmatrix} \hat{u}_m \\ \hat{v}_m \\ \hat{h}_m \end{pmatrix} e^{ik_m x}, \tag{15.2}$$

where $k_m = \frac{2\pi}{L_m}$, L_m is the wavelength and $(\hat{u}_m, \hat{v}_m, \hat{h}_m)$ are the amplitudes of wavenumber m.

If we now substitute (15.2) into (15.1a)–(15.1c) and evoke the principle permitting the superposition of the solutions for the various wavenumbers, then we obtain the matrix vector equation

$$\frac{\partial}{\partial t}\begin{pmatrix} \hat{u}_m \\ \hat{v}_m \\ \hat{h}_m \end{pmatrix} + \begin{pmatrix} G_u & -f & ik_m g \\ f & G_v & 0 \\ ik_m H & 0 & G_h \end{pmatrix}\begin{pmatrix} \hat{u}_m \\ \hat{v}_m \\ \hat{h}_m \end{pmatrix} = \begin{pmatrix} G_u\left(\hat{u}_m\right)_{obs} \\ G_v\left(\hat{v}_m\right)_{obs} \\ G_h\left(\hat{h}_m\right)_{obs} \end{pmatrix}, \tag{15.3}$$

for wavenumber m. It is possible to express (15.3) in terms of matrices-vector notation as

$$\frac{\partial \boldsymbol{z}}{dt} + \mathbf{A}\boldsymbol{z} = \boldsymbol{b}. \tag{15.4}$$

From differential equation theory, we know that the complementary function that solves the homogeneous version of (15.1c) is given by

$$\boldsymbol{z} = \boldsymbol{c}_1 e^{-\lambda_1 t} + \boldsymbol{c}_2 e^{-\lambda_2 t} + \boldsymbol{c}_3 e^{-\lambda_3 t}, \tag{15.5}$$

where the $\boldsymbol{c}_i$, for $i = 1, 2, 3$, are constant vectors that depend of the matrix $\mathbf{A}$ and the initial conditions. The particular integral z_p is given by $z_p = \mathbf{A}^{-1}\boldsymbol{b}$ for constant matrix $\mathbf{A}$ and vector $\boldsymbol{b}$. The λ_i, $i = 1, 2, 3$ are the eigenvalues of $\mathbf{A}$ and are obtained as solutions to

$$\lambda^2 - \lambda^2 (G_u + G_v + G_h) + \lambda \left(G_u G_v + G_v G_h + G_u G_h + k_m^2 g H + f^2 \right) - \left(G_u G_v G_h + G_v k_m^2 g H + G_h f^2 \right) = 0. \tag{15.6}$$

It is stated in [178] that the normal response for this system of equations consists of one stationary and two oscillating error modes, all of which are damped in time. The oscillating waves resemble the gravitational, inertial, or inertia-gravitational modes of the system when dynamic initialization is not included. The damping rates and the frequency of the oscillating waves are functions of the horizontal and vertical scales, the Coriolis parameter, and the magnitudes of the nudging G factors. The three response modes are created by the initial conditions that are neither in geostrophic balance nor equal to the nudging observations.

Given that the solutions of the system include the nudging terms are now functions of these nudging terms, there are many different combinations for these coefficient to affect the system. In [178] the authors consider three different combinations for the nudging terms that represent different physical situations. These situations assume values for G, f, and H of 10^{-3} s^{-1}, 10^{-4} s^{-1}, and 1000 m respectively, and are:

(1) nudging to the rotational component v of the wind field only $\Rightarrow G_v = G$, $G_u = G_h = 0$;
(2) nudging both the rotational v and divergent components $u \Rightarrow G_u = G_v = G$ $G_h = 0$; and
(3) nudging h alone $\Rightarrow G_u = G_v = 0$, $G_h = G$.

In [178] the authors go on to consider a two-dimensional model to approximate the jet stream with incorrect temperature initial conditions and show that the nudging technique could overcome this initial error. The conclusion from [178] is that dynamic initialization through using a nudging technique is capable of causing unbalanced inexact initial first guess of mass and momentum fields to reach satisfactorily accurate, balanced states in different models.

While this technique is not used for global numerical weather prediction, it is useful in determining the model's sensitivity to changes in initial conditions.

A form of nudging, referred to as **spectral nudging**, is used in climate models and also in reanalyses. The aim of this type of nudging is to perturb the prognostic variables of a model ψ_m toward the corresponding variables, ψ_h, in a host model. The standard Newtonian relaxation is applied by taking the difference between ψ_m and ψ_h, such that $\Delta\psi = \psi_m - \psi_h$, and then using this to correct the model such that ψ_m tends to $\psi_m - \alpha\Delta\psi$, where $\alpha \in [0, 1]$, determines the strength of the nudging.

Spectral nudging extends the Newtonian relation method through taking the correcting term and applying a spectral, or low pass, filter so that large spatial wavelengths are adjusted while smaller wavelengths are left unperturbed. A more detailed application of spectral modeling can be found in [444,490].

In [461] the authors apply spectral nudging for the Weather, Research and Forecasting (WRF) model to the tendencies of the variables that relax a selected part of the spectrum to the corresponding waves from the reanalysis. Mathematically, spectral nudging is defined as

$$\frac{dQ}{dt} = L(Q) - \sum_{|n| \le N} \sum_{|m| \le M} K \left(Q_{mn} - Q_{d_{mn}} \right) e^{ik_m x} e^{ik_n y}, \tag{15.7}$$

where Q is any prognostic variable that is to be nudged, L is the model operator, Q_d is the variable from what [461] mention as the *driving fields*, but where Q_{mn} and $Q_{d_{nm}}$ are spectral coefficients.

15.3 Successive Correction

This is the last of the empirical-based data assimilation schemes, although when the theory for this approach was being derived, it was not referred to as such. The first instance of what would be coined **successive correction** appears in [79], which was based upon the initial work by [32]. In this section we shall summarize the work of Bergthórsson and Döös [32], Cressman and Barnes, whose theories are still used for certain applications today and were used for operational numerical weather prediction and ocean prediction in the 1970s and 1980s [16], along with optimum interpolation, which we shall introduce in the next section. We start by summarizing the motivations and derivations from Bergthórsson and Döös [32].

15.3.1 Bergthórsson and Döös [32]

An interesting feature of Bergthórsson and Döös's [32] paper is the description of how numerical weather prediction began to be undertaken in the 1950s. Here are the first two paragraphs of [32]:

> The first attempts at numerical weather forecasting on a routine basis have been characterized by a combination of tedious manual work on one hand and electronic computations with extremely high speed on the other. The weather observations are plotted on maps, examined and analyzed. From this manual analysis values are interpolated at a great number of grid points and punched on a paper copied. Finally the electronic computer can start the forecasting procedure. The manual part of these operations consumes time that is out of proportion to the time required for the machine computation. This, however, is not the only disadvantage.
>
> The manual analyst cannot be expected to use systematic and quantitative methods in his interpolations and extrapolations. His work is rather a complicated curve-fitting by the eye based on a number of more or less well established rules. The analysis will, in other words, be subjective and depending on the skill of the meteorologist. It is furthermore very difficult to avoid wiggles and irregularities of small scale which are neither desirable nor justified by observations. These may frequently amplify in the forecast computation and thus reduce the value of the final forecast. Errors in the reading and punching of values in grid points are also highly probable.

The starting point for the derivation of what would become successive correction in [79] and [32] was to use the observations of the wind and height fields at the 500 mb height as the "informations," along with the 12- or 24-hour barotropic forecast valid for the same time as the analysis, and the normal height of the 500 mb level for the particular month when the analysis is made.

The "analysis" was produced as follows: the authors started with the best available approximation of the 500 mb map; this preliminary field was then modified "as far as possible" with available observations. The analysis obtained could then be used as a preliminary field, that could be modified by observations.

The first preliminary height field Z_p was constructed as a weighted mean of the forecast heights, Z_f, and the normal heights, Z_N, where a normal height is more commonly now referred to as a climatological height. For each grid point, the promilitary height field is defined as

$$Z_p \equiv \frac{\mu_f Z_f + \mu_N Z_N}{\mu_f + \mu_N}, \tag{15.8}$$

where μ_f and μ_N are the weights of the forecast and normal heights, respectively. It is then assumed that the forecast weight is only a function of geographical position and season, as well as the assumption that the deviations from normals are not correlated with the deviations from the forecast. Therefore,

$$\mu_f = \frac{\text{const}}{\sigma f}, \tag{15.9}$$

which is to say that the weight of the forecast is inversely proportional to the root mean square of the differences between observed and forecast heights. By the same argument the weight of the normal at each point is

$$\mu_N = \frac{\text{const}}{\sigma_N}, \tag{15.10}$$

where σ_N is the root mean square of the deviation of the daily values Z from Z_N.

In [32] the authors state that they consider the 500 mb height and wind observations from each station that is less than 900 km away from the grid point and they assume that there are three different approximate height field values:

(1) Assume that the difference between the observed height Z_{os} and the preliminary value at the station Z_{ps} is the same as the difference between the derived height Z_1 and the preliminary height, Z_{pg} at the grid point, which leads to

$$Z_1 = Z_{pg} + \left(Z_{os} - Z_{ps}\right). \tag{15.11}$$

(2) Assuming that the observed wind is in geostrophic balance, and that it is representative of between the grid point and the station, then it is possible to compute the corresponding gradient of Z, which results in a second approximation for the height value given by

$$Z_2 =_{os} + \left(\frac{\partial Z}{\partial n}\right)_{os} \cdot l, \tag{15.12}$$

where l is the distance between the station and the grid point.

(3) We now assume the gradient of the preliminary field at the grid point to be representative for the area between the station and the grid point, therefore the height at this point will be

$$Z_3 = Z_{os} + \left(\frac{\partial Z}{\partial n}\right)_{pg} \cdot l. \tag{15.13}$$

The next step in [32] is quite similar to steps that are undertaken in modern-day variational data assimilation methods, where the authors assume that the weights of the three heights, Z_1, Z_2, and Z_3, are functions of distance between the grid points, where these weights are determined statistically through forming a regression equation. We shall go into more detail about regression later in this chapter, but suffice to say here the regression equation used in [32] is

$$Z_g = \frac{\mu_1 Z_1 + \mu_2 Z_2 + \mu 3 Z_3 + \mu_f Z_f}{\mu_1 + \mu_2 + \mu_3 + \mu_f}. \tag{15.14}$$

However—and here is where the empirical component of this approach comes in—the weights μ_i for $i, 2, 3, f$ are not determined from a statistical regression but from a series of subjectively analyzed charts of the 500 mb height. The specific formulas were

$$\mu_1 = \frac{30}{r^4 + 150} - 0.04, \tag{15.15a}$$

$$\mu_2 = \mu_3 = \frac{27}{r^8 + 70}, \tag{15.15b}$$

$$\mu_h = \frac{2.25}{r^8 + 5} + \frac{10}{r^4 + 20} - 0.01, \tag{15.15c}$$

where μ_h is the weighting for where the station only reports the height field, and as such we have

$$Z_h = Z_{pg} + \left(Z_{os} - Z_{ps}\right). \tag{15.16}$$

It is stated in [32] that the authors found it to be of little use to apply the height Z_h from stations beyond 1500 km. We have a copy of figure 3 from [32] in Fig. 15.1.

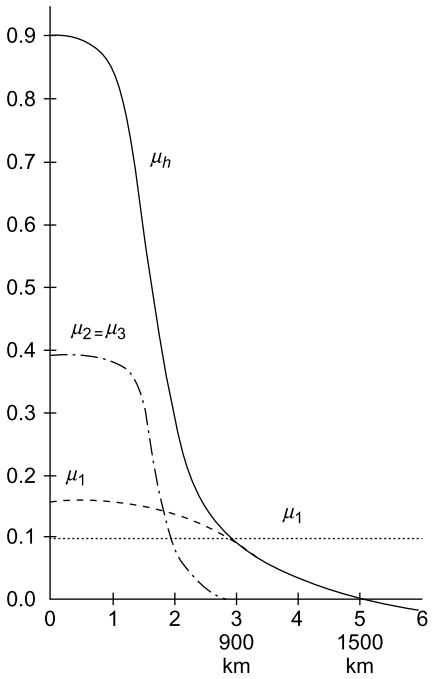


FIGURE 15.1

Copy of figure 3 from Páll Bergthörsson and Bo R. Döös (1955), Numerical Weather Map Analysis, Tellus, 7:3, 329-340, DOI: 10.3402/tellusa.v7i3.8902, https://creativecommons.org/licenses/by/4.0/.

The expression for the value of the height at the grid point, where the nearby station only reported height, is given by

$$Z_g = \frac{\mu_f Z_f + \mu_N Z_N + \mu_h Z_h}{\mu_f + \mu_N + \mu_h}. \tag{15.17}$$

In the case of several stations influencing the height at the point, another factor has to be taken into account. Some stations may be very close to each other and therefore give contributions that are correlated, while other stations are isolated and give contributions that are more or less uncorrelated to those of other stations. The weight of the contributions from a station should therefore be reduced [32]. For this reduction the authors introduce a factor that is inversely proportional to the density of stations surrounding the particular station, $\frac{1}{\varrho}$. In [32] the authors evaluate ϱ as the number of stations within a radius of 375 km. This leads to the expression for the height at the grid point as

$$Zg = \frac{\mu_f Z_f + \mu_N Z_N + \sum_{i=1}^{n} \frac{1}{\varrho_i} (\mu_1 Z_1 + \mu_2 Z_2 + \mu_3 Z_3)_i + \sum_{j=1}^{m} \frac{1}{\varrho_j} (\mu_h Z_h)_j}{\mu_f + \mu_N + \sum_{i=1}^{n} \frac{1}{\varrho_i} (\mu_1 + \mu_2 + \mu_3)_i + \sum_{j=1}^{m} \frac{1}{\varrho_j} (\mu_h)_j}. \tag{15.18}$$

15.3.2 Cressman [79]

We now consider the slight variation of the method of Bergthórsson and Döös from Cressman [79], and has been used operationally in the United States national weather service in the past. It is in [79] that we see the term *successive correction*, but the scheme is not called that here.

We are again looking at height and wind data at different stations that may or may not be at the location of the numerical grid being considered. There is a stereographic plot of the domain in [79]. There it is stated that if only height observations are reported, then the correction C_h is computed from a nearby grid point by

$$C_h = -W E_h, \tag{15.19}$$

where E_h is the error of the interpolated value of the first guess field at the location of the observation, and W is the weighting factor given by

$$W = \frac{N^2 - d^2}{N^2 + d^2}, \tag{15.20}$$

where d is the distance between the grid point and the observation location, and N is the distance that W goes to zero. The term N is also referred to as the **radius of influence** and can be altered on any iteration. The advantage of being able to change the radius of influence is that it enables different scales in the atmosphere to be resolved on each iteration.

We should note that the expression in (15.20) is a simplified version of the weights from [32] where the reasoning for this choice for the weights was to minimize the amount of computing required. Note that this is still a priority today in operational environments.

If both a height and a wind are reported at a station, then the correction, C_v, is computed for a nearby grid point as

$$C_v = W\left(D_o + \frac{kf}{mg}(v\Delta x - u\Delta y) - D_g\right), \tag{15.21}$$

where D_0 is the observed height at the location of the observation, u and v are the observed wind components, m is the map scale factor, Δx and Δy are the component of the map distance from the observing point to the grid point, and D_g is the value of the height at the grid point in the first guess. The k term is the average ratio of the geostrophic wind to the actual wind and is set to 1.08. This amounts to passing a plane in the xyz coordinates through the observation, multiplying the increment between the plane and the first guess of the pressure surface W, and adding the resulting increment to the first guess.

If only a wind is reported, then C_v is determined by (15.21), except that D_o is the height interpolated from the first guess at the location of the observation.

All the data within the radius N of the grid point are under consideration. Each observation reporting a height is used in the computation of a value of C_h, and each observation reporting a wind is used in computing a value C_v. Therefore, for a station reporting both height and wind fields, we have both C_h and C_V as we saw in [32].

Once all of the values of the C_hs and the C_vs have been computed, then the correction C to be applied is determined as a weighted mean by

$$C = \frac{A\sum_{h,i}^{n_h} C_h + \sum_{j=1}^{n_v} C_v}{An_h + n_v}, \tag{15.22}$$

where n_h is the number of C_h values and n_v is the number of C_v values, and A is a weighting factor. In [79] the weighting factor A is suggested as being the weight given to the lateral gradient of the first guess as compared to the observed winds, which have a weight of 1. It is stated in [79] that the fit to the observed winds was desired and as such a value $A = 2^{-5}$ is applied in the results shown in [79].

An important difference between the methods from [32] and [79] is that there are multiple iterations, while in [32] there is only one or two; note that the iterations in [79] are referred to as **scans**. The reason for this terminology is due to the procedure for this scheme. For each scan the value of N is successively reduced. Therefore, on each scan we have a correction that has been computed for a grid point from the data with a circle of radius N are averaged to obtain a single point correction from (15.22), which means that there has been a type of smoothing that has taken place over this circle. The use of a series of scans with decreasing radii allows the analysis of a spectrum of scales. This implies that the largest value of N used can be set to permit correction of the largest scale errors in the first guess, while the smallest value sets a lower limit to the scale that can be analyzed.

The technique from [79] was used for many different applications in many geophysical fields and was used in the 1970s for the assimilation of aircraft data in tropical cyclones [31]. In [31] the authors use objective analysis to assimilate aircraft observations in tropical cyclones. In their motivation they indicate that they required their weighting functions to have a high degree of circular symmetry so as to match that of the characteristics of tropical cyclones. Their weighting functions are given by

$$w_{ijk} = \frac{D_{ij}^2 \sin\phi + \left(D_{ij}^2 - d_{ijk}^2\right)\cos\phi}{Dij^2 + d_{ijk}^2}, \quad \text{if } \left(r_k - r_{ij}\right)^2 < D_{ij}^2$$

$$wijk = 0, \quad \text{if } \left(r_k - r_{ij}\right)^2 \geq D_{ij}^2,$$

where $D_{ij}^2 = hr_{ij}$ and $d_{ijk}^2 = (r_k - rij)^2 + r_{i,j}^2 \left(\theta_k - \theta_{ij}\right)^2$. See [31] for an illustration of the coordinate system they use for the objective analysis scheme.

An interesting spin on the Cressman scheme appears in [216], where the authors attempt to introduce wind corrections in a four-dimensional data assimilation framework. Their assimilation technique was based upon assimilating asynoptic height fields that had been derived from Vertical Temperature Profile Radiometers (VTPR) soundings between 0600 UTC and 1800 UTC. An important detail in [216] is the cycling of the model *forward and backward*, where the backward cycling refers to applying what is called the highly damping Euler-backward time integration [230], which is designed to suppress any *shocks* created during the data assimilation process. But that the height fields were introduced to the model grid point through the Cressman successive correction scheme. However, the "insertion" technique refers to introducing a wind correction that comprised of the predicted ageostrophic component, and the analyzed geostrophic component, along with a localized height, at specific times, with or without localized balancing [169,305].

15.3.3 Barnes [24]

The motivation for the approach derived in [24] was because Barnes felt that the other techniques being used at the time were surface fitting schemes, which are characterized as fitting a geometric surface to a reported data and calculating the values determined by that surface at any other points of interest, specifically the grid points. Barnes states that he felt that there were three disadvantages to these types of approaches:

- The calculations are complicated and require considerable time to complete.
- The data to which the surface is fitted are chosen in a rather artificial manner, that which produces the best results.
- The effects of erroneous data can be disastrous since each datum is given equal ranking in determining the shape of the surface.

Barnes does mention the smoothing process that Cressman had introduced, where he refers to the methods of Bergthórsson and Döös and Cressman as weighted-averaging, but highlights that the major disadvantage of these methods is that they had a tendency to smooth out all small variations in the fields, whether caused by data errors, or actual atmospheric disturbances.

Barnes states in [24] that the weight factors used in his objective analysis have been developed from the fundamental premise that the two-dimensional distribution of an atmospheric variable can be represented by the summation of an infinite number of *independent harmonic waves*, that is to say by a Fourier integral representation.

The development of the interpolation scheme from [24] is as follows: under the assumption that the distribution of an atmospheric quantity, $f(x, y)$, can be depicted by a Fourier integral representation, what we refer to nowadays as a Fourier transform, then it is possible to define a corresponding smoothed function $g(x, y)$, which is obtained by applying a filter to the original functions, as

$$g(x, y) = \int_0^{2\pi} \int_0^{\infty} f(x + r\cos\theta, y + r\sin\theta)\, w dr d\theta, \tag{15.23}$$

with the weight factor, or filter, is

$$w = \frac{1}{4\pi k} e^{-\frac{r^2}{4k}}, \tag{15.24}$$

where r and θ are polar coordinates, the origin being at the point (x, y), and k is a parameter determining the shape of the weight factors which is related to the density of the observed data concerning $f(x, y)$.

The next step was to rearrange (15.23) in order to express the weight factor in an alternative form:

$$g(x, y) = \int_0^{2\pi} \int_0^{\infty} f(x + r\cos\theta, y + r\sin\theta) \left(\frac{\eta}{2\pi}\right) d\left(\frac{r^2}{4k}\right) d\theta, \tag{15.25}$$

where

$$r = e^{-\frac{r^2}{4k}}, \tag{15.26}$$

is the new weight factor. Barnes preferred the formulation in (15.25) compared to (15.23) because in the latter, the maximum weight is not applied at $r = 0$.

However, interpolation using (15.25) is not practical because firstly we do not know the analytical form of $f(x, y)$, and secondly it is not possible to integrate that function to infinity. Therefore, $g(x, y)$ has to be approximated through placing a finite limit on the region of influence of any datum concerning $f(x, y)$ and through taking a weighted average of only those M number of data within that region. This implied that

$$g(x, y) = \frac{\sum_j^M \eta(\tau_j) \cdot f_j}{\sum_{j=1}^M \eta(r_j)}. \tag{15.27}$$

A straightforward property that we can see from (15.22) is

$$\int_0^{2\pi} \int_0^{2\pi} \frac{1}{2\pi} \eta d\left(\frac{r^2}{4k}\right) d\theta = 1.$$

Integrating with respect to θ and to some distance R, it is possible to write the integral above as

$$\int_0^R \eta d\left(\frac{r^2}{4k}\right) + \int_R^{\infty} \eta d\left(\frac{r^2}{4k}\right) = 1.$$

The second integral above is defined as ε, which implies

$$\int_0^R e^{-\frac{r^2}{4k}} d\left(\frac{r^2}{4k}\right) = 1 - \varepsilon.$$

Applying the integration to the expression above yields

$$e^{-\frac{R^2}{4k}}, \text{ or } \frac{R^2}{4k} = -\ln\varepsilon = E. \tag{15.28}$$

If ε is small enough, then it is possible to represent the weighted influence of any datum with sufficient accuracy. In [24] Barnes says that if $E = 4$, then that means that the scheme has represented

98% of the influence of any datum within the circular region whose radius is R, where R is the **radius of influence** of the weight factor η.

An interesting feature about the Barnes successive correction scheme compared to the two earlier schemes in this section is that it does not have an initial state; also the weighting function presented for the three schemes are quite different.

We now move on to the theory of least squares and nonlinear least squares estimation that play an important role in data assimilation.

15.4 Linear and Nonlinear Least Squares

In this section we shall introduce the techniques to derive and solve the linear and nonlinear weighted least squares problem. We start with a linear univariate weighted least squares problem.

15.4.1 Univariate Linear Least Squares

The best way to introduce least squares, and weighted least squares, is through a toy problem. A classic example, which is used in many classes/lectures on least squares, is to consider the case where you are recording a temperature of a situation, either in a room, or at a specific location. You start with a prior estimate of the temperature, referred to here as T_b, and that you have an observation of that temperature, denoted T_o. There will also exist a true temperature that we do not know but is denoted T_t. We can define the *errors* in the background and the observation as

$$\varepsilon_b = T_b - T_t,$$
$$\varepsilon_o = T_o - T_t.$$

We now assume that the errors above are *unbiased*; this implies $\overline{\varepsilon_b} = \overline{\varepsilon_o} = 0$, where the bar refers to the mean of the errors.

The next step is to form an *analysis*, which is a linear combination of the background and the observed value of the temperature. The analysis temperature is denoted by T_a and is defined as

$$T_a = \alpha_1 T_o + \alpha_2 T_b + \alpha_3, \tag{15.29}$$

where α_3 is a constant. We now define the *analysis error* as $\varepsilon_a = T_a - T_t$, where we require this error to be unbiased, which implies $\overline{\varepsilon_a} = 0$. If we now express the analysis state in terms of the true state and the background and observation errors, then we have

$$T_a = T_t + \varepsilon_a = \alpha_1 (T_t + \varepsilon_b) + \alpha_2 (T_t + \varepsilon_o) + \alpha_3. \tag{15.30}$$

If we now take the expectation of (15.30), which is also referred to as taking the mean, we see that using the fact that the mean of the background and observation errors have been assumed to be zero, then we obtain an expression for the mean analysis error, $\overline{\varepsilon_a}$, as

$$\overline{\varepsilon_a} = (\alpha_1 + \alpha_2 - 1) T_t + \alpha_3 = 0. \tag{15.31}$$

Since (15.31) must hold for all true temperature values, including the case where $T_t = 0$, then $\alpha_3 = 0$. This implies that $\alpha_1 + \alpha_2 = 1$, and $\alpha_2 = 1 - \alpha_1$. We shall now drop the subscripts on α as there is only one remaining. Therefore the **general linear unbiased estimate** for this problem is

$$T_a = \alpha T_o + (1 - \alpha) T_b. \tag{15.32}$$

The next step is to determine what the value of α is. To achieve this we consider the error of the estimate in (15.32). If we now subtract the true state from both sides of the equation in (15.32), and recall the definitions for the various errors, then

$$\varepsilon_a = \alpha \varepsilon_o + (1 - \alpha) \varepsilon_b. \tag{15.33}$$

If we now form the variance of (15.33), which we shall denote by σ_i^2 for $i = a, b, o$, then

$$\sigma_a^2 = \alpha^2 \sigma_o^2 + 2\alpha (1 - \alpha) \sigma_o \sigma_b + (1 - \alpha)^2 \sigma_b^2, \tag{15.34}$$

where we have used the property

$$VAR[aX_1 + bX_2] = a^2 VAR[X_1] + b^2 VAR[X_2] + 2abCOV[X_1 X_2].$$

We now assume no covariance between the background error and the observation error, which leaves

$$\sigma_a^2 = \alpha^2 \sigma_o^2 + (1 - \alpha) \sigma_b^2. \tag{15.35}$$

We now wish to consider three properties of the estimate in (15.35). If we take the first derivative of (15.35) with respect to α, then

$$\frac{d\sigma_a^2}{d\alpha} = 2\alpha \sigma_o^2 - 2(1 - \alpha) \sigma_b^2. \tag{15.36}$$

If we consider the case where $\alpha = 0$, then $\sigma_a^2 = \sigma_b^2$ and from (15.36) $\frac{d\sigma_a^2}{d\alpha} = -2\sigma_b^2 < 0$. If we now consider the case when $\alpha = 1$, then $\sigma_a^2 = \sigma_o^2$, and from (15.36) we have $\frac{d\sigma_a^2}{d\alpha} = 2\sigma_o^2 > 0$.

Given this information about $\alpha = 0$ and $\alpha = 1$, we can deduce that for $0 \leq \alpha \leq 1$, then the analysis variance is less than or equal to the maximum of the background or observation variance, that is to say, $\sigma_a^2 \leq \max\left(\sigma_b^2, \sigma_0^2\right)$. We also have that the minimum variance estimate occurs for a value of α that lies between 0 and 1 but not including them, that is to say, for $\alpha \in (0, 1)$. Finally, we have that the minimum variance estimate satisfies

$$\sigma_a^2 < \min\left(\sigma_b^2, \sigma_o^2\right).$$

Therefore, the minimum variance estimate occurs when the derivative in (15.36) is equal to zero,

$$\frac{d\sigma_a^2}{d\alpha} = 2\alpha \sigma_o^2 - 2(1 - \alpha) \sigma_b^2 = 0, \quad \Rightarrow \quad \alpha = \frac{\sigma_b^2}{\sigma_o^2 + \sigma_b^2}. \tag{15.37}$$

Thus, the error variance of this now **minimum variance** estimate is

$$\sigma_a^2 = \left(\frac{1}{\sigma_b^2} + \frac{1}{\sigma_o^2}\right)^{-1}. \tag{15.38}$$

Therefore, the estimate with value of α derived in (15.37) is referred to as the **best linear unbiased estimate**, or **BLUE**.

To help illustrate this structure of the analysis we have plotted a simple one-dimensional case where we consider the temperature in a room where we have background temperature of $21c$ and an observation of $22c$ where the background variance is given by 1.5 and the observational variance is 1 in Fig. 15.2. We can see that the analysis distribution is between the background and observation's Gaussian PDFs and that the analysis state has a higher probability.

Exercise 15.1. *For the situation described above find the value of the analysis state as well as the analysis variance.*

15.4.2 Multidimensional Least Squares

We now move on to the multiple dimension problem where instead of a scalar prior estimate, where we have a vector, denoted $\boldsymbol{x}_b$, with b referring to the **background**. This vector could be viewed as representing the complete state of a numerical model at some time, where the elements of $\boldsymbol{x}_b$ may be grid point values, spherical harmonic coefficients, but also where the vector's elements do not represent only one physical attribute, i.e., not just temperature but temperature, winds, sea surface temperature, soil moisture, salinity, magma density, etc.

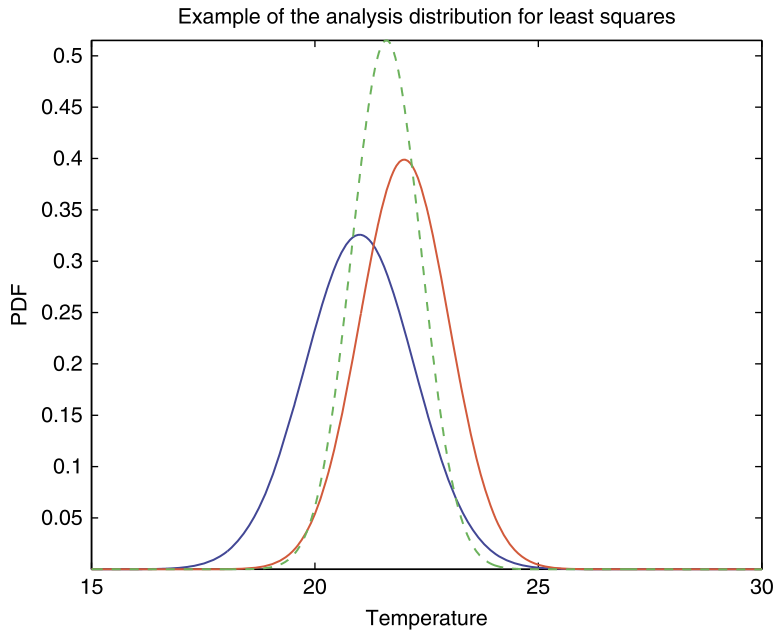


FIGURE 15.2

Plot of the background (blue), observational (red), and analysis (green) distributions from a least squares approach.

We now introduce the vector of observations $\boldsymbol{y}$, that contains both direct and indirect observations, at different locations to the grid points, and of different variables. Therefore, we need to be able to match the background state to these locations and in terms of these observations. To formalize this we introduce the **observation/forward operator**, which can be a nonlinear operator, and is denoted by $\boldsymbol{h}(\boldsymbol{x})$. This means that $\boldsymbol{h}(\boldsymbol{x})$ can be compared to $\boldsymbol{y}$, where the $\boldsymbol{h}(\boldsymbol{x})$ represents the **model equivalent** of $\boldsymbol{y}$.

For the time being we are going to assume that the observation operator does not introduce any errors, so that

$$\boldsymbol{h}(\boldsymbol{x}_t) = \boldsymbol{y}_t, \tag{15.39}$$

where $\boldsymbol{x}_t$ is vector of the true states, and $\boldsymbol{y}$ contains the true values of the observed quantities.

As with the scalar case, we seek an analysis that is a linear combination of the background and the observations, which in a matrix vector form is

$$\boldsymbol{x}_a = \mathbf{F}\boldsymbol{x}_b + \mathbf{G}\boldsymbol{h}(\boldsymbol{x}_b) + \boldsymbol{K}\boldsymbol{y} + \boldsymbol{c}, \tag{15.40}$$

where $\mathbf{F}$, $\mathbf{G}$, and $\mathbf{K}$ are matrices to be determined, and $\mathbf{c}$ is an unknown vector.

If we assume that $\boldsymbol{h}$ is linear, then we can look for a linear unbiased estimate as we did for the scalar case. If, however, $\boldsymbol{h}$ is nonlinear, then we require that error-free inputs, that is to say, $\boldsymbol{x}_b = \boldsymbol{x}_t$ and $\boldsymbol{y} = \boldsymbol{y}_t$, produce error-free analysis, that is to say, $\boldsymbol{x}_a = \boldsymbol{x}_t$, which then implies

$$\boldsymbol{x}_t = \mathbf{F}\boldsymbol{x}_t + \mathbf{G}\boldsymbol{h}(\boldsymbol{x}_t) + \mathbf{K}\boldsymbol{h}(\boldsymbol{x}_t) + \boldsymbol{c}. \tag{15.41}$$

We require the expression in (15.41) to hold for all true states, including the case where $\boldsymbol{x}_t = \mathbf{0}$, which then implies that $\boldsymbol{c} = \mathbf{0}$. This in turn implies that

$$\mathbf{F}\boldsymbol{x}_t + \mathbf{G}\boldsymbol{h}(\boldsymbol{x}_t) = \mathbf{I}\boldsymbol{x}_t - \mathbf{K}\boldsymbol{h}(\boldsymbol{x}_t), \tag{15.42}$$

and therefore we can see that the expression in (15.42) can only hold if $\mathbf{F} = \mathbf{I}$ and $\mathbf{G} = -\mathbf{K}$. Therefore, the analysis equation for this situation is

$$\boldsymbol{x}_a = \boldsymbol{x}_b + \mathbf{K}(\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_b)). \tag{15.43}$$

We now need to determine what the matrix $\mathbf{K}$ is. If we recall from the scalar case we had that

$$T_a = \alpha T_o + (1-\alpha) T_b, \equiv T_b + \alpha (T_o - T_b),$$

which is similar in appearance to the matrix equation in (15.43), and therefore, the $\mathbf{K}$ matrix plays a role of the weights given to the observations as well as handling the information of the transformations between the **observation space** and **model space**. The matrix $\boldsymbol{K}$ is referred to as the **gain matrix**.

We now introduce the definition of the errors for the multi-dimensional analysis error, along with the multidimensional versions of the background and observation errors as

$$\boldsymbol{\varepsilon}_a \equiv \boldsymbol{x}_a - \boldsymbol{x}_t, \tag{15.44a}$$
$$\boldsymbol{\varepsilon}_b \equiv \boldsymbol{x}_b - \boldsymbol{x}_t, \tag{15.44b}$$
$$\boldsymbol{\varepsilon}_o \equiv \boldsymbol{y} - \boldsymbol{y}_t. \tag{15.44c}$$

The next step is to make the assumption that the errors are small, which enables us to linearize the observation operator about the background state, and have a linear model for the error. This implies

$$\boldsymbol{h}(\boldsymbol{x}_b) = \boldsymbol{h}(\boldsymbol{t}_b) + \mathbf{H}\boldsymbol{\varepsilon}_b + \mathcal{O}\left(\boldsymbol{\varepsilon}_b^2\right), \tag{15.45}$$

for $\mathbf{H}$ is the Jacobian of the observation operator, defined as

$$\mathbf{H}_{ij} = \frac{\partial h_i(\boldsymbol{x}_b)}{\partial x_j}, \tag{15.46}$$

where $i = 1, 2, \ldots, N_o$ and $j = 1, 2, \ldots, N$, where N_o is the number of observations and N is the number of entries in $\boldsymbol{x}$.

We now substitute the expressions for the errors from (15.44a) to (15.44c) into the analysis equation and use the property that $\boldsymbol{h}(\boldsymbol{x}_t) = \boldsymbol{y}_t$, which results in the following equation to the first order of

$$\boldsymbol{\varepsilon}_a = \boldsymbol{\varepsilon}_b + \mathbf{K}(\boldsymbol{\varepsilon}_o - \mathbf{H}\boldsymbol{\varepsilon}_b). \tag{15.47}$$

We now assume that the mean errors have be removed, so that $\mathbb{E}[\boldsymbol{\varepsilon}_b] = \mathbb{E}[\boldsymbol{\varepsilon}_o] = \mathbf{0}$; given this information, we see from (15.47) that $\mathbb{E}[\boldsymbol{\varepsilon}_a] = 0$.

As we are now seeking a statistical weight, we recall the definition for the covariance and expectation operator for a multivariate distribution,

$$\mathbb{C} = \mathbb{E}\left[(x_i - \overline{x}_i)\left(x_j - \overline{x}_j\right)\right], \tag{15.48}$$

as well as the property that covariances matrices are symmetric and positive definite.

We return to the analysis error equation (15.47), which we can rewrite in the following form,

$$\boldsymbol{\varepsilon}_a = (\mathbf{I} - \mathbf{KH})\boldsymbol{\varepsilon}_b + \mathbf{K}\boldsymbol{\varepsilon}_o. \tag{15.49}$$

The next step is to form the **analysis error covariance matrix** which is obtained by taking the expectation of the product of the vector of analysis error with its transpose, which is referred to as an **outer product**. Therefore, we have

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\varepsilon}_a\boldsymbol{\varepsilon}_a^T\right] &= \mathbb{E}\left[\left((\mathbf{I} - \mathbf{KH})\boldsymbol{\varepsilon}_b + \mathbf{K}\boldsymbol{\varepsilon}_o\right)\left((\mathbf{I} - \mathbf{KH})\boldsymbol{\varepsilon}_b + \mathbf{K}\boldsymbol{\varepsilon}_o\right)^T\right], \\
&= (\mathbf{I} - \mathbf{KH})\,\mathbb{E}\left[\boldsymbol{\varepsilon}_b\boldsymbol{\varepsilon}_b^T\right](\mathbf{I} - \mathbf{KH})^T + (\mathbf{I} - \mathbf{KH})\,\mathbb{E}\left[\boldsymbol{\varepsilon}_b\boldsymbol{\varepsilon}_o^T\right]\mathbf{K}^T \\
&\quad + \mathbf{K}\mathbb{E}\left[\boldsymbol{\varepsilon}_o\boldsymbol{\varepsilon}_b^T\right](\mathbf{I} - \mathbf{KH})^T + \mathbf{K}\mathbb{E}\left[\boldsymbol{\varepsilon}_o\boldsymbol{\varepsilon}_o^T\right]\mathbf{K}^T.
\end{aligned} \tag{15.50}$$

As with the scalar case, we assume that no correlation between the background errors and the observational errors. As a result (15.50) simplifies to

$$\mathbb{E}\left[\boldsymbol{\varepsilon}_a\boldsymbol{\varepsilon}_a^T\right] = (\mathbf{I} - \mathbf{KH})\,\mathbb{E}\left[\boldsymbol{\varepsilon}_b\boldsymbol{\varepsilon}_b^T\right](\mathbf{I} - \mathbf{KH})^T + \mathbf{K}\mathbb{E}\left[\boldsymbol{\varepsilon}_o\boldsymbol{\varepsilon}_o^T\right]\mathbf{K}^T, \tag{15.51}$$

where we can see that the expression in (15.51) is similar to the scalar version we derived earlier, i.e., $\sigma_a^2 = (1-\alpha)^2\sigma_b^2 + \alpha^2\sigma_o^2$. Thus, we see that the $\mathbf{K}$ matrix in (15.51) is equivalent to α in the scalar case, where we showed that the value of α was chosen so as to minimize the variance.

However, we have to address how we minimize the variance in a multidimensional situation. We have seen from Chapter 4 that the covariance matrices contain the variances of the variable in their

diagonal entries. This implies that it is possible to define the minimum variance analysis as that which minimized the sum of the diagonal elements of the analysis error covariance matrix. It is possible to obtain a function of the diagonal entries of a matrix through its *trace*, which we shall denote as Tr. We recall that for the scalar case we obtain the expression for the minimum variance analysis by setting $\frac{d\sigma_a^2}{d\alpha} = 0$. For the multidimensional case we do something similar, but with respect to the trace of the analysis covariance matrix as

$$\frac{\mathrm{Tr}\left(\mathbb{E}\left[\boldsymbol{\varepsilon}_a \boldsymbol{\varepsilon}_a^T\right]\right)}{\partial \mathbf{K}} = \mathbf{0}. \tag{15.52}$$

In order to apply the derivative in (15.52) to the expression we have for the analysis error covariance matrix in (15.50), we require the following identities:

$$\begin{aligned}
\frac{\mathrm{Tr}\left(\mathbf{K}\mathbf{A}\mathbf{K}^T\right)}{\partial \mathbf{K}} &= \mathbf{K}^T\left(\mathbf{A} + \mathbf{A}^T\right), \\
\frac{\mathrm{Tr}\left(\mathbf{K}\mathbf{A}\right)}{\partial \mathbf{K}} &= \mathbf{A}^T, \\
\frac{\mathrm{Tr}\left(\mathbf{A}\mathbf{K}^T\right)}{\partial \mathbf{K}} &= \mathbf{A}.
\end{aligned}$$

Substituting these properties into (15.52) acting on (15.50) results in

$$\frac{\mathrm{Tr}\left(\mathbb{E}\left[\boldsymbol{\varepsilon}_a \boldsymbol{\varepsilon}_a^T\right]\right)}{\partial \mathbf{K}} = 2\mathbf{K}\left(\mathbf{H}\mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right]\mathbf{H}^T + \mathbb{E}\left[\boldsymbol{\varepsilon}_o \boldsymbol{\varepsilon}_o^T\right]\right) - 2\mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right]\mathbf{H}^T = \mathbf{0}. \tag{15.53}$$

Rearranging (15.53) and factorizing leads to the expression for $\mathbf{K}$ as

$$\mathbf{K} = \mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right]\mathbf{H}^T\left(\mathbf{H}\mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right]\mathbf{H}^T + \mathbb{E}\left[\boldsymbol{\varepsilon}_o \boldsymbol{\varepsilon}_o^T\right]\right)^{-1}. \tag{15.54}$$

The expression in (15.54) is for the optimal gain matrix which is called the **Kalman gain matrix**; we shall see why in Chapter 19. We can see that the appearance of the optimal expression for $\mathbf{K}$ in (15.54) is quite similar in appearance to that of the scalar case, where we have $\alpha = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_o^2}$.

Exercise 15.2. *Show that the analysis covariance for the scalar case is*

$$\sigma_a^2 = \left(\frac{1}{\sigma_b^2} + \frac{1}{\sigma_0^2}\right)^{-1},$$

and that the analysis variance matrix is

$$\mathbb{E}\left[\boldsymbol{\varepsilon}_a \boldsymbol{\varepsilon}_a^T\right] = \left(\mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right]^{-1} + \mathbf{H}^T\mathbb{E}\left[\boldsymbol{\varepsilon}_o \boldsymbol{\varepsilon}_o^T\right]^{-1}\mathbf{H}\right)^{-1}.$$

The method that we have used to obtain the expressions for the minimum variance estimates is the **weighted least sum of squares** approach. If we had not had the α or the $\mathbf{K}$ terms, then that would be the method of **least sum of squares**.

Before we move on to nonlinear least squares theory, we introduce some standardized notation for the different covariance matrices in (15.54) as set out in [190]:

$$\mathbf{P}^a \equiv \mathbb{E}\left[\boldsymbol{\varepsilon}_a \boldsymbol{\varepsilon}_a^T\right], \quad \mathbf{P}^b \equiv \mathbb{E}\left[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T\right], \quad \mathbf{R} = \mathbb{E}\left[\boldsymbol{\varepsilon}_o \boldsymbol{\varepsilon}_o^T\right], \tag{15.55}$$

where $\mathbf{P}^a$ is the **analysis error covariance matrix**, $\mathbf{P}^b$ is the **background error variance matrix** and $\mathbf{R}$ is the **observation error covariance matrix**. **Note:** in most variational data assimilation derivation the background error covariance is usually approximated and is denoted by $\mathbf{B}$.

Exercise 15.3. *Show that the Kalman gain matrix can also be defined as*

$$\mathbf{K} = \left(\left(\mathbf{P}^b\right)^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{R}^{-1}.$$

15.4.3 Nonlinear Least Squares Theory

The derivation that we show in this section come from two papers in the early 1980s, which derive some interesting functionals and functions that will appear similar to some of the equations involved in the variational-based data assimilation. The two papers we are going to summarize are Tarantola and Valette [430,431].

The opening sentence in [431] states that:

> The aim of physical sciences is to discover the minimal set of parameters which completely describes physical systems and the laws relating the values of these parameters to the results of any set of measurements on the system.

Given this aim, Tarantola and Valette start the motivation by introducing the loose definition of direct/forward and inverse problems. These problems are defined as follows.

Definition 15.4. If given some information on the values of the set of parameter, we try to use a theoretical relationship in order to obtain information on the values of some measurable quantity; this is solving a **direct/forward** problem.

If given some information on the values of some measured quantities, we try to use a theoretical relationship in order to obtain information on the values of the set of parameters; this is solving an **inverse** problem [431].

As a caveat, Tarantola and Valette state that one of the difficulties arising in the solution of some problems is the instability (a small change in the inputs of the problems produces a physically unacceptable large change in the outputs), where this difficulty arises in both direct and inverse problems. However, inverse problems have the extra difficulty of **non-uniqueness**. Tarantola and Valette state that there are two reasons for non-uniqueness; the first comes from the fact that the data are discrete; if the data were dense, then the solution would be unique, as proven in Backus and Gilbert [18]. The second reason for non-uniqueness is explained through a brief example of the inverse problem where you are obtaining the density structure of a region of the Earth from the measurement of the local gravitational field: Gauss' theorem states that an infinity of different density configurations gives identical gravitational fields.

Another example of this can be associated with brightness temperature when a cloud is present and we are trying to invert to obtain the hydrometeors along with the synoptic variables. If there is not a

cloud in the first guess, then there is no sensitivity to the cloud variables and hence we can also obtain the same brightness temperature by adjusting the temperature and wind fields [391].

Returning to [431], the authors introduce the following notation: let $\mathscr{L}$ be a large physical system that comprises of a physical system and the measuring instruments. It is said that $\mathscr{L}$ is *parameterizable* if any state of $\mathscr{L}$ maybe described using some functions and some discrete parameters. This then implies that $\mathscr{L}$ is quantitative. If any state of $\mathscr{L}$ may be described using a finite set of discrete parameters, that it is said that $\mathscr{L}$ is a *discrete system*.

Given a discrete system, we let $\boldsymbol{X} = \{X_1, X_2, \ldots, X_m\}$ be the finite set of parameters needed to describe the system, the particular values of the parameters are denoted as $\boldsymbol{x} = \{x_1, x_2, \ldots, x_m\}$. As we are including all of the measuring instruments in $\mathscr{L}$, the parameter set contains all the data and the unknowns of the problem. The next step in [431] is to define the m-dimensional space where the parameters, $\boldsymbol{X}$, take their values; this space is denoted by $\mathscr{E}^m$ and as such $\boldsymbol{x}$ is a point in that space which is referred to as the **state**.

We now introduce constraints between the possible values of the parameters, where in a functional form are

$$\begin{aligned} f_1(x_1, x_2, \ldots, x_m) &= 0, \\ f_2(x_1, x_2, \ldots, x_m) &= 0, \\ &\vdots \\ f_r(x_1, x_2, \ldots, x_m) &= 0, \end{aligned} \tag{15.56}$$

and in vectorial form;

$$\boldsymbol{f}(\boldsymbol{x}) = 0. \tag{15.57}$$

It is quite common to be able to partition the sets of parameters as

$$\boldsymbol{X} = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_m \end{pmatrix} = \begin{pmatrix} D_1 \\ D_2 \\ \vdots \\ D_r \\ P_1 \\ P_2 \\ \vdots \\ P_s \end{pmatrix} = \begin{pmatrix} \mathbf{D} \\ \mathbf{P} \end{pmatrix}, \tag{15.58}$$

where $\mathbf{D}$ is the set of data and $\mathbf{P}$ is the set of unknowns. This enables us to simplify (15.56) to

$$\begin{aligned} d_1 &= g_1(x_1, x_2, \ldots, x_m), \\ d_2 &= g_2(x_1, x_2, \ldots, x_m), \\ &\vdots \\ d_r &= g_r(x_1, x_2, \ldots, x_m), \end{aligned} \tag{15.59}$$

which again can be written in vectorial form as

$$\boldsymbol{d} = \boldsymbol{g}(\boldsymbol{x}). \tag{15.60}$$

We now consider the inputs to the problem. We select a specific parameter, X_α, where there are two possibilities that can arise in that X_α is either directly measurable or it is not. This leads to the following two important statements in [431]:

> If X_α is a directly measurable parameter, and that it has been measured, then we assume that the results of the measurement has a Gaussian form; this implies that it may be conveniently described using the expected value $x_{\alpha,0}$, along with the variance and covariances with other measurements.
> If X_α is not a directly measurable parameter, which implies that it is an unknown, then we assume, in order to be able to solve what would be an underdetermined problem, that we have some **a priori** knowledge and that this information may also be expressed in a Gaussian form. If the a priori information about a parameter is weak, not very accurate, then the corresponding variance will be large, of even could be infinite.

The a priori information may come from different sources, such as from a numerical model, or it could be posteriori information of a previous inverse problem run with different data sets. However, according to [431] it is often the case that the a priori information is obtained through placing *reasonable* error bars around a *reasonable* central values [196]. This then provides stability and uniqueness to the inversion of the problem.

Therefore, in the least squares approach it is assumed that all a priori information on the parameter set, which includes both the measurable and non-measurable parameters, takes the form of a **vector of expected values**, $\boldsymbol{x}_0$, and a **covariance matrix**, $\mathbf{C}_0$. Therefore, given the expression in (15.60), we have that it is possible to partition the vector of expected values and the covariance matrix as

$$\boldsymbol{x}_0 \equiv \begin{pmatrix} \boldsymbol{d}_0 \\ \boldsymbol{p}_0 \end{pmatrix}, \quad \mathbf{C}_0 \equiv \begin{pmatrix} \mathbf{C}_{d_0 d_0} & \mathbf{C}_{d_0 p_0} \\ \mathbf{C}_{p_0 d_0} & \mathbf{C}_{p_0 p_0} \end{pmatrix}, \tag{15.61}$$

where we shall assume that the covariances between the observations and the parameters are equal to zero, that is to say, $\mathbf{B}_{d_0 p_0} = \mathbf{B}^T_{p_0 d_0} = \mathbf{0}$.

Addressing now the least squares problem associated with the assumption of Gaussianity for the a priori information, we have that $\boldsymbol{x}_0$ and $\mathbf{C}_0$ define a Gaussian probability density function in the parameter space, $\mathscr{E}_m$, given by

$$\rho(\boldsymbol{x}) = c \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_0)^T \mathbf{C}_0^{-1}(\boldsymbol{x}-\boldsymbol{x}_0)\right\}. \tag{15.62}$$

We recall that for the **nonlinear** theoretical equations defined in (15.57);

$$\mathcal{M}(\boldsymbol{x}) = \mathbf{0},$$

defines a nonlinear manifold of subspace in $\mathscr{E}_m$, denoted by $\mathscr{F}$ in [431]. It is stated in [431] that the priori density function, $\rho(\boldsymbol{x})$, induces a PDF on the theoretical manifold $\mathscr{F}$, that could be taken as the posteriori density function.

An important point that Tarantola and Valette make at this juncture is that if the numerical model is **linear**, then $\mathscr{F}$ will be a linear manifold, and in [430] it is shown that the induced probability density

function will generally be Gaussian; however, if the numerical model is **not** linear then the induced PDF will, in general, **not** be Gaussian.

In [431] the authors state that the least squares problem is the search for the point $\widehat{\boldsymbol{x}}$ of the theoretical manifold for which the induced density of probability is maximum. To find the maximum just stated, we seek the minimum of the argument in the exponential in (15.62).

Thus the nonlinear least squares problem is we seek a point, $\widehat{\boldsymbol{x}}$, that verifies the set of equations:

$$\mathcal{M}(\widehat{\boldsymbol{x}}) = 0, \tag{15.63}$$

$$s(\widehat{\boldsymbol{x}}) = (\boldsymbol{x} - \boldsymbol{x}_0)^T \mathbf{C}_0^{-1} (\boldsymbol{x} - \boldsymbol{x}_0), \text{ such that } s(\widehat{\boldsymbol{x}}) \text{ is minimum over } \mathscr{F}. \tag{15.64}$$

However, if a point $\widehat{\boldsymbol{x}}$ verifies (15.63) and (15.64), then it must also verify the set of equations

$$\mathcal{M}(\widehat{\boldsymbol{x}}) = 0, \tag{15.65}$$

$$s(\widehat{\boldsymbol{x}}) \text{ stationary over } \mathscr{F}. \tag{15.66}$$

A caveat to note here is that in the problem of searching for minima, (15.65) and (15.66) may contain, in addition to the solutions of (15.63) and (15.64), local minima, saddle points, and maxima.

We now derive a nonlinear equation for the solution to (15.65) and (15.66). We start by recalling the definition of the tangent linear model of the nonlinear model $\mathcal{M}(\boldsymbol{x})$ as

$$\mathbf{M}_{ij} \equiv \frac{\partial \mathcal{M}_i}{\partial x_j}, \tag{15.67}$$

and is assumed to be full rank. Tarantola and Valette refer to (15.67) as the *tangent linear application*, and apply the tangent linear application to s, denoted by S, which results in

$$S_j \equiv \frac{\partial s}{\partial x_j}. \tag{15.68}$$

Next, let $\widehat{\boldsymbol{x}}$ be a solution of (15.65) and (15.66), then s is stationary at $\widehat{\boldsymbol{x}}$, which implies that S is equal to zero over the tangent linear manifold to $\mathscr{F}$ at $\widehat{\boldsymbol{x}}$, and as such

$$S = 2(\widehat{\boldsymbol{x}} - \boldsymbol{x}_0)^T \mathbf{C}_0^{-1}. \tag{15.69}$$

We introduce a new vector, $\boldsymbol{v}$, that belongs to the tangent linear manifold to $\mathscr{F}$ at $\widehat{\boldsymbol{x}}$, if and only if, $\mathbf{M}\boldsymbol{v} = \mathbf{0}$. This implies that it is possible to write (15.65) and (15.66) as

$$\mathcal{M}(\widehat{\boldsymbol{x}}) = \mathbf{0}, \tag{15.70}$$

$$\mathbf{M}\boldsymbol{v} = \mathbf{0} \Rightarrow (\widehat{\boldsymbol{x}} - \boldsymbol{x}_0)^T \mathbf{C}_0^{-1} \boldsymbol{v} = \mathbf{0}. \tag{15.71}$$

As we have assumed that the tangent linear matrix has full rank, this implies that there exists a vector of Lagrange multipliers, $\boldsymbol{\lambda}$, such that $(\widehat{\boldsymbol{x}} - \boldsymbol{x}_0)^T \mathbf{C}_0^{-1} \boldsymbol{v} = \boldsymbol{\lambda}^T \mathbf{M}$. This then enables us to write (15.70) and (15.71) as

$$\mathcal{M}(\widehat{\boldsymbol{x}}) = \mathbf{0}, \tag{15.72a}$$

$$\exists \boldsymbol{\lambda} : (\widehat{\boldsymbol{x}} - \boldsymbol{x}_0) = \mathbf{C}_0 \mathbf{M}^T \boldsymbol{\lambda}. \tag{15.72b}$$

If we now multiply on the left of (15.72b) by $\mathbf{M}$, then we obtain

$$\mathbf{M}\boldsymbol{\lambda} : (\widehat{\boldsymbol{x}} - \boldsymbol{x}_0) = \left(\mathbf{M}\mathbf{C}_0\mathbf{M}^T\right)\boldsymbol{\lambda}.$$

Due to the positive definiteness of the covariance matrix, and the full rank of the tangent linear matrix, we obtain the following expression for the Lagrange multiplier as

$$\boldsymbol{\lambda} = \left(\mathbf{M}\mathbf{C}_0\mathbf{M}^T\right)^{-1}\mathbf{M}\left(\widehat{\boldsymbol{x}} - \boldsymbol{x}_0\right). \tag{15.73}$$

Given the expression for the vector of Lagrange multipliers in (15.73), we can write (15.65) and (15.66) as the following nonlinear single equation:

$$\widehat{\boldsymbol{x}} = \boldsymbol{x}_0 + \mathbf{C}_0\mathbf{M}^T\left(\mathbf{M}\mathbf{C}_0\mathbf{M}^T\right)^{-1}\left(\mathbf{M}\left(\widehat{\boldsymbol{x}} - \boldsymbol{x}_0\right) - \mathcal{M}\left(\widehat{\boldsymbol{x}}\right)\right). \tag{15.74}$$

For us to be able to find solutions to (15.74) we are required to apply an iterative solver of some sorts. Tarantola and Valette now assume that the elements of the tangent linear matrix are continuous functions of $\boldsymbol{x}$, then as a result of this assumption then the simplest procedure to solve (15.74) is carried out by using a **fixed point** method, which assumes that the derivatives are evaluated at the k iterate, $\boldsymbol{x}_k$, which can be written as

$$\widehat{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_0 + \mathbf{C}_0\mathbf{M}_k^T\left(\mathbf{M}_k\mathbf{C}_0\mathbf{M}_k^T\right)^{-1}\left(\mathbf{M}_k\left(\widehat{\boldsymbol{x}}_k - \boldsymbol{x}_0\right) - \mathcal{M}\left(\widehat{\boldsymbol{x}}_k\right)\right). \tag{15.75}$$

The algorithm defined in (15.75) is referred to in [431] as the algorithm of **Total Inversion (TI)**.

If we now assume that the covariance matrix $\mathbf{C}_0$ does not contain null (zero) variances, nor does it have perfect correlations, and if we assume that (15.57) can be written in the form (15.60), then we can partition $\mathbf{F}$ into the form

$$\mathbf{F} = \begin{pmatrix}\mathbf{I} & -\mathbf{G}\end{pmatrix}, \tag{15.76}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{G}$ is a matrix of partial derivative of the function $\boldsymbol{g}$.

We now consider the case of $\boldsymbol{d} = \boldsymbol{g}(\boldsymbol{p})$, where we assume that the parameters, $\boldsymbol{X}$, may be divided into the data set, $\mathbf{D}$, and parameter set, $\mathbf{P}$, such that the theoretical equations, $\mathcal{M}(\boldsymbol{x}) = \mathbf{0}$, and they simplify to

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{d} - \boldsymbol{g}(\boldsymbol{p}) = \mathbf{0}. \tag{15.77}$$

Given the partition of the tangent linear matrix as presented in (15.76) where

$$G_{ij} = \frac{\partial g_i}{\partial p_j}, \tag{15.78}$$

then through using (15.61), (15.77), and (15.76), we obtain for the k iteration:

$$\begin{aligned}\mathbf{M}_k\left(\boldsymbol{x}_k - \boldsymbol{x}_0\right) - \boldsymbol{f}(\boldsymbol{x}) &= \begin{pmatrix}\mathbf{I} & \mathbf{G}_k\end{pmatrix}\begin{pmatrix}\widehat{\boldsymbol{d}}_k - \boldsymbol{d}_0 \\ \widehat{\boldsymbol{p}}_k - \boldsymbol{p}_0\end{pmatrix} - \left(\widehat{\boldsymbol{d}}_k - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right)\right), \\ &= -\left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k\left(\widehat{\boldsymbol{p}}_k - \boldsymbol{p}_0\right)\right),\end{aligned} \tag{15.79}$$

$$\mathbf{C}_0\mathbf{F}_k^T = \begin{pmatrix} \mathbf{C}_{d_0d_0} & \mathbf{C}_{d_0p_0} \\ \mathbf{C}_{p_0d_0} & \mathbf{C}_{p_0p_0} \end{pmatrix}\begin{pmatrix} \mathbf{I} \\ -\mathbf{G}_k^T \end{pmatrix},$$

$$= \begin{pmatrix} \mathbf{C}_{d_0d_0} - \mathbf{C}_{d_0p_0}\mathbf{G}_k^T \\ \mathbf{C}_{p_0d_0} - \mathbf{C}_{p_0p_0}\mathbf{G}_k^T \end{pmatrix}, \tag{15.80}$$

$$\mathbf{F}_k\mathbf{C}_0\mathbf{F}_k^T = \mathbf{C}_{d_0d_0} - \mathbf{C}_{d_0p_0}\mathbf{G}_k^T - \mathbf{G}_k\mathbf{C}_{p_0d_0} + \mathbf{G}_k\mathbf{C}_{p_0p_0}\mathbf{G}_k^T, \tag{15.81}$$

and upon substituting (15.79)–(15.81) into (15.75), we obtain

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_0 + \left(\mathbf{C}_{p_0p_0}\mathbf{G}_k^T - \mathbf{C}_{p_0d_0}\right)\left(\mathbf{C}_{d_0d_0} - \mathbf{C}_{d_0p_0}\mathbf{G}_k^T - \mathbf{G}_k\mathbf{C}_{p_0d_0} + \mathbf{G}_k\mathbf{C}_{p_0p_0}\mathbf{G}_k^T\right)^{-1}\left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k\left(\widehat{\boldsymbol{p}}_k - \boldsymbol{p}_0\right)\right). \tag{15.82}$$

The corresponding algorithm for the iterations for the data is given by

$$\widehat{\boldsymbol{d}}_{k+1} = \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k\left(\widehat{\boldsymbol{p}}_{k+1} - \widehat{\boldsymbol{p}}_k\right). \tag{15.83}$$

However, we assume in data assimilation that the correlations between the background and the observation errors are zero. The reason for using the data assimilation terminology is to show that with this assumption (15.82) and (15.83) are the equations for the nonlinear solution to incremental 3D VAR cost functions, but the background state is updated each time and so are the Jacobians of the observation operators. Therefore, (15.82), in terms of data assimilation notation, becomes

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_b + \left(\mathbf{B}\mathbf{H}_k^T\right)\left(\mathbf{R} + \mathbf{H}_k\mathbf{B}\mathbf{H}_k^T\right)^{-1}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_k\right) + \mathbf{H}_k\delta\boldsymbol{x}\right), \tag{15.84}$$

where $\delta\boldsymbol{x} \equiv \boldsymbol{x}_k - \boldsymbol{x}_b$, and $\mathbf{C}_{p_0p_0} \equiv \mathbf{B}$, $\mathbf{C}_{d_0d_0} \equiv \mathbf{R}$, $\mathbf{G} \equiv \mathbf{H}$ and $\boldsymbol{g}\left(\boldsymbol{p}\right) \equiv \boldsymbol{h}\left(\boldsymbol{x}\right)$; however, we have substitute the actual "data," what we now call observations, $\boldsymbol{y}$ for the mean state of the data $\boldsymbol{d}_0$. Therefore, the significance of Eq. (20) in Tarantola and Valette [430] is that a nonlinear solution solver for the 3D VAR cost function existed many years before atmospheric scientists would derive similar equations [259], but also that (15.84) can be rearranged to be a version of the linear solution from [259], which in itself would become the equivalent to the solution of incremental 3D VAR.

To illustrate this further, Tarantola and Valette present two more solution that are rearrangements of the solution in (15.82), which are Eqs. (24) and (25) in [431]

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_0 + \mathbf{C}_{p_0p_0}\mathbf{G}_k^T\left(\mathbf{C}_{d_0d_0} + \mathbf{G}_k\mathbf{C}_{p_0p_0}\mathbf{G}_k^T\right)^{-1}\left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k\left(\widehat{\boldsymbol{p}}_{k+1} - \boldsymbol{p}_0\right)\right), \tag{15.85a}$$

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_0 + \left(\mathbf{G}_k^T\mathbf{C}_{d_0d_0}^{-1}\mathbf{G}_k + \mathbf{C}_{p_0p_0}^{-1}\right)^{-1}\mathbf{G}_k^T\mathbf{C}_{d_0d_0}^{-1}\left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k\left(\widehat{\boldsymbol{p}}_{k+1} - \boldsymbol{p}_0\right)\right), \tag{15.85b}$$

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_k + \left(\mathbf{G}_k^T\mathbf{C}_{d_0d_0}^{-1}\mathbf{G}_k + \mathbf{C}_{p_0p_0}^{-1}\right)^{-1}\left(\mathbf{G}_k^T\mathbf{C}_{d_0d_0}^{-1}\left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right)\right) - \mathbf{C}_{p_0p_0}^{-1}\left(\widehat{\boldsymbol{p}}_k - \boldsymbol{p}_0\right)\right). \tag{15.85c}$$

Using data assimilation terminology again we can rewrite (15.85a)–(15.85c) as

$$\widehat{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_b + \mathbf{B}\mathbf{H}_k^T\left(\mathbf{R} + \mathbf{H}_k\mathbf{B}\mathbf{H}_k^T\right)^{-1}\left(\boldsymbol{y} - \boldsymbol{h}\left(\widehat{\boldsymbol{x}}_k\right) + \mathbf{H}_k\delta\boldsymbol{x}_k\right), \tag{15.86a}$$

$$\widehat{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_b + \left(\mathbf{H}_k^T\mathbf{R}^{-1}\mathbf{H}_k + \mathbf{B}^{-1}\right)^{-1}\mathbf{H}_k^T\mathbf{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{h}\left(\widehat{\boldsymbol{x}}_k\right) + \mathbf{H}_k\delta\boldsymbol{x}_k\right), \tag{15.86b}$$

$$\widehat{\boldsymbol{x}}_{k+1} = \boldsymbol{x}_k + \left(\mathbf{H}_k^T \mathbf{R}^{-1} \mathbf{H}_k + \mathbf{B}^{-1}\right)^{-1} \left(\mathbf{H}_k^T \mathbf{R} \left(\boldsymbol{y} - \boldsymbol{h}\left(\widehat{\boldsymbol{x}}_k\right)\right) - \mathbf{B}^{-1} \delta \widehat{\boldsymbol{x}}\right). \tag{15.86c}$$

The proofs of (15.85a)–(15.85c) can be found in [431].

It is not just the fully nonlinear incremental version of 3D VAR that is derived in [431], but also optimum interpolation, which we shall introduce in the next section. However, in [430] the cost function that will later be known as that of 3D VAR [259] was presented through a **non-Bayesian** approach; Tarantola and Veletta's reason for this is that *it is too restrictive for our purposes*. The theory that is presented in [430] is based upon conjunction of states of information.

The reason for introducing this theory, and we have one more to go from [431], is to make the reader aware that theoretical development can quite often be in parallel in different geoscience fields, and that the need for interdisciplinary research, but also interdisciplinary sharing of research, is important. As the author I will admit I had never heard of the Tarantola and Valette papers until I was researching optimum interpolation and stumbled across a paper title "*Optimal Control Theory Applied to an Objective Analysis of a Tidal Current Mapping by HF Radar*," [97], which happened to be in a copy of a journal that I was looking for a paper on the National Oceanic and Atmospheric Administration (NOAA)'s operational ocean thermal analysis system [63], where [430,431] were cited.

Before we move on to linear regression, we present a final caveat from [431]; although it is not presented as such, it should remind us to be careful when linearizing nonlinear problems.

Section 2f of Tarantola and Valette (1982b) starts with the statement:

> The usual approach to solving the nonlinear problem is through iteration of a linearized problem. If the data set **D** overdetermines the problem sufficiently so that all the a priori information on the parameter set **P** can be neglected, then the iteration of a linearized problem always leads to the correct solution. If the data set does not overdetermine the problem, there is a common mistake which leads to a wrong solution.

We have the solution to the nonlinear problem given by (15.85c),

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_k + \left(\mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \mathbf{G}_k + \mathbf{C}_{p_0 p_0}^{-1}\right)^{-1} \left(\mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right)\right) + \mathbf{C}_{p_0 p_0}^{-1} \left(\boldsymbol{p}_0 - \widehat{\boldsymbol{p}}_k\right)\right),$$

which can be shown to be equivalent to

$$\widehat{\boldsymbol{p}}_{k+1} = \boldsymbol{p}_0 + \left(\mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \mathbf{G}_k + \mathbf{C}_{p_0 p_0}^{-1}\right)^{-1} \mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right) + \mathbf{G}_k \left(\widehat{\boldsymbol{p}}_{k+1} - \boldsymbol{p}_0\right)\right).$$

We have not presented the derivation of the solution to the linear problem in this subsection, but it was presented in the linear least squares subsection; however, in keeping with the notation from [431], the linear solution is given by

$$\widehat{\boldsymbol{p}} = \boldsymbol{p}_0 + \left(\mathbf{G}^T \mathbf{C}_{d_0 d_0}^{-1} \mathbf{G} + \mathbf{C}_{p_0 p_0}^{-1}\right)^{-1} \mathbf{G}^T \mathbf{C}_{d_0 d_0}^{-1} \left(\boldsymbol{d}_0 - \mathbf{G}\boldsymbol{p}_0\right). \tag{15.87}$$

Therefore, if we require the linearization approach to be consistent, then we must force the general nonlinear solution to give the linear solution as a particular case. To linearize the problem means replacing the nonlinear equation, $\boldsymbol{d} = \boldsymbol{g}(\boldsymbol{p})$, by a Taylor series approximation about the point $\boldsymbol{p}_k$ as

$$\boldsymbol{d} = \boldsymbol{g}\left(\boldsymbol{p}_k\right)\left(\boldsymbol{p} - \boldsymbol{p}_k\right).$$

The next expression we require for the linearization is **residuals**. In [431] they are defined as

$$\Delta\widehat{\boldsymbol{d}}_k = \boldsymbol{d}_0 - \boldsymbol{g}\left(\boldsymbol{p}_k\right), \tag{15.88}$$

which defines the **corrections** as

$$\Delta\widehat{\boldsymbol{p}}_{k+1} = \widehat{\boldsymbol{p}}_{k+1} - \widehat{\boldsymbol{p}}_k. \tag{15.89}$$

Given the definitions for the residual and the correction, we can express the linearized least squares problem in terms of the search for the values of $\Delta\boldsymbol{p}_{k+1}$ that minimize the sum

$$s = \left(\mathbf{G}_k \Delta\widehat{\boldsymbol{p}}_{k+1} - \Delta\widehat{\boldsymbol{d}}_k\right)^T \mathbf{C}_{d_0 d_0}^{-1} \left(\mathbf{G}_k \Delta\widehat{\boldsymbol{p}}_{k+1} - \Delta\widehat{\boldsymbol{d}}_k\right), \tag{15.90}$$

if the problem is overdetermined enough. For an underdetermined problem it is usually required that each successive correction, $\Delta\widehat{\boldsymbol{p}}_{k+1}$, be as small as possible; this leads to the sum in (15.90) being replaced by

$$s' = s + \left(\Delta\widehat{\boldsymbol{p}}_{k+1}\right)^T \mathbf{C}_{p_0 p_0}^{-1} \left(\Delta\widehat{\boldsymbol{p}}_{k+1}\right). \tag{15.91}$$

The corresponding solution can be found to be

$$\Delta\widehat{\boldsymbol{p}}_{k+1} = \left(\mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \mathbf{G}_k + \mathbf{C}_{p_0 p_0}^{-1}\right)^{-1} \mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right)\right). \tag{15.92}$$

By using (15.88) and (15.89), the algorithm for (15.92) is given by

$$\widehat{\boldsymbol{p}}_{k+1} = \widehat{\boldsymbol{p}}_k + \left(\mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \mathbf{G}_k + \mathbf{C}_{p_0 p_0}^{-1}\right)^{-1} \mathbf{G}_k^T \mathbf{C}_{d_0 d_0}^{-1} \left(\boldsymbol{d}_0 - \boldsymbol{g}\left(\widehat{\boldsymbol{p}}_k\right)\right). \tag{15.93}$$

However, upon comparing (15.93) with (15.85c), we see that the $\mathbf{C}_{p_0 p_0}^{-1}\left(\boldsymbol{p}_0 - \boldsymbol{p}_k\right)$ is missing. This means that the linear solution in (15.87) cannot be obtained from (15.93), which implies that (15.93) is wrong. However, to correct this problem we simply need to change the condition that each successive **partial correction**, $\widehat{\boldsymbol{p}}_{k+1} - \widehat{\boldsymbol{p}}_k$, be small, where in fact we actually require that the **total correction**, $\widehat{\boldsymbol{p}}_{k+1} - \boldsymbol{p}_0$ be as small as possible. Replacing the definition for the correction to be the total correction results in (15.85c).

A final note about the work presented in [431] is in relation to the "general nonlinear least squares problem" section, where we have the situation of a system that is described through a set, $\boldsymbol{X}$, of continuous, and or discrete, vectors. Let $\boldsymbol{x}_0$ be the a priori value of $\boldsymbol{X}$, and let $\mathbf{C}_0$ be the corresponding covariance operator. If we now let a physical theory impose a nonlinear relationship of the form

$$\boldsymbol{f}(\boldsymbol{x}) = \mathbf{0}, \tag{15.94}$$

on the possible values of $\boldsymbol{X}$, where $\boldsymbol{f}$ is any nonlinear differentiable operator acting on $\boldsymbol{X}$, then $\boldsymbol{x}_0$ **does not have to verify** (15.94).

We define the least squares problem as that of the search for the point $\widehat{\boldsymbol{x}}$ minimizing

$$s(\boldsymbol{x}) = \langle (\widehat{\boldsymbol{x}} - \boldsymbol{x}_0)^T, \mathbf{C}_0^{-1} (\widehat{\boldsymbol{x}} - \boldsymbol{x}_0) \rangle, \tag{15.95}$$

among the points that verify (15.94), where $\langle \cdot, \cdot \rangle$ represents the inner product.

The solution of (15.95) combined with the constraint of (15.94) can be shown to be

$$\widehat{\boldsymbol{x}} = \boldsymbol{x}_0 + \mathbf{C}_0^{-1}\mathbf{F}^* \left(\mathbf{F}\mathbf{C}_0^{-1}\mathbf{F}^*\right)^{-1} \left(\mathbf{F}\left(\widehat{\boldsymbol{x}} - \boldsymbol{x}_0\right) - \boldsymbol{f}\left(\widehat{\boldsymbol{x}}\right)\right), \tag{15.96}$$

where the linear operator $\mathbf{F}$ is the derivatives of the nonlinear operator $\boldsymbol{f}$, and where $\mathbf{F}^*$ is its **adjoint**.

The remainder of [431] examines first the theory of Barkus and Gilbert [18] and shows that it is a specific case of their nonlinear least squares theory and then moves on to defining different covariance/correlation functions. Both [430] and [431] are worth a read, as they effectively derive 3D VAR without noticing the significance of the derivation. An important feature of [430] is that the authors almost could have derived 4D VAR if they had not integrated off the time component in their toy example.

15.5 Regression

We now move on to an application of the least squares theory that is used in many different forms of data assimilation. The starting point is to assume that we have a set of data points, observations, (x_i, y_i), and where, for the first form of regression we present, we assume that the particular x_i, y_i of two variables X and Y are related by the equations

$$Y_i = \beta x_i + \varepsilon, \tag{15.97}$$

where β is the gradient of the underlying straight-line relationship, and ε is the error, or residual, of the determining of y_i.

An important feature to note here is that X is fixed equal to x_i, and therefore has no error attached to it, which makes X a mathematical variable, and not a random variable; however, Y is a random variable. In addition to depending on X, there is also a random error that plays a role in determining Y, so that the values of y_i corresponds to a given fixed value of x_i. There are two components that define this relationship between Y and x_i. The first of these is the βx_i term which accounts for the straight-line relationship between y_i and x_i. The second component is the random error, ε_i, which is added to take into account the residual in determining y_i.

The residual terms, ε_i, $i = 1, 2, \ldots, N$, are assumed to be independent random variables from a distribution with mean equal to zero and with variance σ^2. It therefore follows that each of the y_is is a random sample from a distribution with mean βx_i and variance σ^2, and that the y_is are independent of each other.

In Fig. 15.3 we have plotted an example of a set of points (x_i, y_i) to illustrate the problem that we are trying to address here. We see points that are clearly showing a linear relationship along with the best fit line describing the relationship between them. We can see that some of the original points are below and some are above the line, and on occasion the line passes through some points. We now turn our attention to how to find the line that best fits the data.

The deviation from a line is equal to $y_i - \beta x_i$, so that the sum of squares of the residuals is given by

$$R = \sum_{i=1}^{N} (y_i - \beta x_i)^2. \tag{15.98}$$

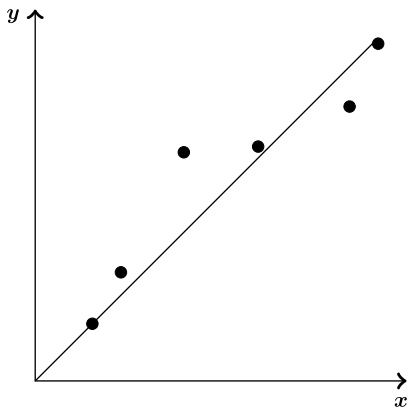


FIGURE 15.3

Simple illustration of a best-line fit through some sample points.

We now seek to minimize R, which is achieved by choosing the appropriate value of β, which will be denoted by b. The slope of the line is allowed to be adjusted to find the best possible fit to the observed set of points (x_i, y_i). As a result of this the values of x_i and y_i are treated as constants in R, so that β is the only variable element in R. Therefore, to find the minimum of R, we consider

$$\frac{dR}{d\beta} = \sum_{i=1}^{N} \left(-2x_i \left(y_i - \beta x_i\right)\right). \tag{15.99}$$

Setting (15.99) equal to zero, we have

$$\sum_{i=1}^{N} x_i \left(y_i - \beta x_i\right) = 0. \tag{15.100}$$

The value of β that satisfies the relationship in (15.100) is b, which leads to

$$\sum_{i=1}^{N} x_i y_i = b \sum_{i=1}^{N} x_i^2 \;\Rightarrow\; b = \frac{\sum_{i=1}^{N} x_i y_i}{\sum_{i=1}^{N} x_i^2}. \tag{15.101}$$

The value of b that is obtained by using N pairs of observed values, x_i, and y_i, is the estimate of the population parameter β found by the method of least squares. We should note that the line fitted here passes through the origin.

We now consider the case where the linear relationship between the sets of points does not pass through the origin; in this case we define the relationship between y and x as

$$y = \alpha + \beta x. \tag{15.102}$$

This means that the observed y_is will be related to the fixed x_is through the equation

$$Y_i = \alpha + \beta x_i + \varepsilon_i. \tag{15.103}$$

The Y_is that correspond to a given x_i will be a random variable with mean $\alpha + \beta x_i$, and whose variance is σ^2. The procedure to fit a linear of the form in (15.103) is again achieved through the method of least squares, but this time we have to find the optimal α and β; however, the estimate for β is slightly more complicated than for the case where the line passes through the origin.

We start by defining the deviations as $y_i - \alpha - \beta x_i$, and as such we wish to find the values for α and β that minimize

$$R = \sum_{i=1}^{N} (y_i - \alpha - \beta x_i)^2 , \tag{15.104}$$

where only α and β can vary.

To derive the optimal estimates of α and β we introduce the change of variables: $p_i \equiv x_i - \overline{x}$ and $q_i \equiv y_i - \overline{y}$, where the bar is the mean value of the variable. This enables us to write the deviation as

$$y_i - \alpha - \beta x_i \equiv (y_i - \overline{y}) - \beta (x_i - \overline{x}) - (\alpha - \overline{y} + \beta \overline{x}) \equiv q_i - \beta p_i - (\alpha - \overline{y} + \beta \overline{x}) , \tag{15.105}$$

and to rewrite (15.104) as

$$R = \sum_{i=1}^{N} (y_i - \alpha - \beta x_i)^2 = \sum_{i=1}^{N} (q_i - \beta p_i - (\alpha - \overline{y} + \beta \overline{x}))^2 . \tag{15.106}$$

The next step is to notice that the cross term in the expansion of the square in (15.106) is

$$-2 \sum_{i=1}^{N} (q_i - \beta p_i) (\alpha - \overline{y} + \beta \overline{x}) = -2 (\alpha - \overline{y} + \beta \overline{x}) \sum_{i=1}^{N} (q_i - \beta p_i) . \tag{15.107}$$

An important property that applies here is that the sums $\sum_{i=1}^{N} p_i$ and $\sum_{i=1}^{N} q_i$ are both equal to zero. This is because of the definitions of the p_is and the q_is being the sum of the differences of the x_is and the y_is about their means, which is zero. This implies that the cross term in the expansion of (15.106) is equal to zero and leaves us with

$$\begin{aligned} R &= \sum_{i=1}^{N} (q_i - \beta p_i)^2 + \sum_{i=1}^{N} (\alpha - \overline{y} + \beta \overline{x})^2 , \\ &= \sum_{i=1}^{N} (q_i - \beta p_i)^2 + N (\alpha - \overline{y} + \beta \overline{x})^2 . \end{aligned} \tag{15.108}$$

The two expression above in (15.108) are either positive, or zero, since each term is proportional to a square, or a sum of squares. Therefore, R will be a minimum when each of the two expressions attains its minimum value. If we take the second expression first and set $\alpha = \overline{y} - \beta \overline{x}$, then the second expression in (15.108) is equal to zero, and as such this must be the minimum as the squared term cannot go negative. The β parameter is still free to be chosen; however, we should recognize that the first term in (15.108) is the same, in form, of the derivation we did in the first part of this section. Therefore, we have that

$$b = \frac{\sum_{i=1}^{N} p_i q_i}{\sum_{i=1}^{N} p_i^2} \equiv \frac{\sum_{i=1}^{N} (x_i - \overline{x}) (y_i - \overline{y})}{\sum_{i=1}^{N} (x_i - \overline{x})^2} . \tag{15.109}$$

15.5.1 Linear Regression Involving Two or More Variables

In most geophysical situations it is highly likely that we wish to fit a linear relationship that is dependent on more than one variable. Suppose that we have two predictor variates X_1 and X_2, with a predicted variate Y, then we wish to fit a linear relationship between the two predictor variates and the predicted variate as

$$Y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \varepsilon. \tag{15.110}$$

However, it could be the case that we wish to fit a relationship of the form

$$Y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon. \tag{15.111}$$

Both (15.110) and (15.111) are examples of linear statistical models as they are linear in terms of the unknown parameters, α, β_1, β_2. Each of these cases are example of multiple linear regression, where regressions describes the relationship of the form $\mathbb{E}[Y|x] = f(x)$, or for multiple variable we would have $\mathbb{E}\left[Y|x_1, x_2, \ldots, x_p\right] = f\left(x_1, x_2, \ldots, x_p\right)$. Note that while X^2 is a higher-order power above linear, it is treated as a second variate.

Therefore we start by considering the model in (15.97), where

$$Y = \alpha + \beta x_1 + \beta x_2 + \varepsilon,$$

where we wish to estimate the parameters α, β_1, and β_2, given a sample of N sets of values (y_i, x_{1i}, x_{2i}) for $i = 1, 2, \ldots, N$.

We again introduce a change of variable that involves the mean of y, x_1 and now x_2, as $p_i \equiv y_i - \overline{y}$, $q_{1i} \equiv x_{1i} - \overline{x}_{1i}$ and $q_{2i} \equiv x_{2i} - \overline{x}_{2i}$. We now rewrite the linear model as

$$V = \hat{\alpha} + \beta_1 q_1 + \beta_2 q_2 + \varepsilon. \tag{15.112}$$

We again peruse the principle of least squares to determine the parameters by minimizing the quantity

$$R = \sum_{i=1}^{N} \left(p_i - \hat{\alpha} - \beta_1 q_{1i} - \beta_2 q_{2i}\right)^2, \tag{15.113}$$

with respect to variations in $\hat{\alpha}$, β_1, and β_2. To achieve this goal we shall differentiate (15.113) with respect to the three parameters, which yields the system of simultaneous equations:

$$\frac{\partial R}{\partial \hat{\alpha}} = -2 \sum_{i=1}^{N} \left(p_i - \hat{a} - b_1 q_{1i} - b_2 q_{2i}\right) = 0, \tag{15.114a}$$

$$\frac{\partial R}{\partial \beta_1} = -2 \sum_{i=1}^{N} q_{1i} \left(p_i - \hat{a} - b_1 q_{1i} - b_2 q_{2i}\right) = 0, \tag{15.114b}$$

$$\frac{\partial R}{\partial \beta_2} = -2 \sum_{i=1}^{N} q_{2i} \left(p_i - \hat{a} - b_1 q_{1i} - b_2 q_{2i}\right) = 0. \tag{15.114c}$$

We can see from (15.114a) that $\hat{\alpha} = 0$ due to the other terms in the summation are equal to zero as they were for the univariate case. Given this information we can now write (15.114b) and (15.114c) as the system of simultaneous equations:

$$b_1 \sum_{i=1}^{N} q_{1i}^2 + b_2 \sum_{i=1}^{N} q_{1i} q_{2i} = \sum_{i=1}^{N} q_{1i} p_i, \tag{15.115a}$$

$$b_1 \sum_{i=1}^{N} q_{1i} q_{2i} + b_2 \sum_{i=1}^{N} q_{2i}^2 = \sum_{i=1}^{N} q_{2i} p_i. \tag{15.115b}$$

The equations given in (15.115a) and (15.115b), and a similar set of simultaneous equations of this form for a higher number of variates, are referred to as the **normal equations**. Note this has nothing to do with the normal distribution. The quantities in the summations are the **corrected sum of squares and products**. We now introduce the following notation for the left-hand terms in (15.115a) and (15.115b) as

$$S_{kl} = \sum_{i=1}^{N} q_{ki} q_{li} = \sum_{i=1}^{N} (x_{ki} - \overline{x}_k)(x_{li} - \overline{x}_l), \text{ for } k = 1, 2; l = 1, 2, \tag{15.116}$$

and the right-hand sides of (15.115a) and (15.115b) as

$$S_{0k} = \sum_{i=1}^{N} q_{ki} p_i = \sum_{i=1}^{N} (x_{ki} - \overline{x}_k)(y_i - \overline{y}), \text{ for } k = 1, 2. \tag{15.117}$$

Given the expressions above in (15.116) and (15.117), we can write (15.115a) and (15.115b) as

$$\begin{aligned} b_1 S_{11} + b_2 S_{12} &= S_{01}, \\ b_1 S_{12} + b_2 S_{22} &= S_{02}. \end{aligned} \tag{15.118}$$

An important thing to note about (15.118) is that it can be written in a matrix-vector equation form as

$$\mathbf{S}\boldsymbol{b} = \mathbf{S}_0, \text{ where } \mathbf{S} \equiv \begin{pmatrix} S_{11} & S_{12} \\ S_{12} & S_{22} \end{pmatrix}, \text{ and } \mathbf{S}_0 \equiv \begin{pmatrix} S_{01} \\ S_{02} \end{pmatrix}, \tag{15.119}$$

and as such the coefficients b_is are found through inverting the matrix $\mathbf{S}$ and multiplying by $\mathbf{S}_0$. Therefore, the expressions for the optimal values for the b_is are

$$b_1 = \frac{S_{22} S_{01} - S_{12} S_{02}}{S_{11} S_{22} - S_{12}^2}, \quad b_2 = \frac{S_{11} S_2 - S_{12} S_{01}}{S_{11} S_{22} - S_{12}^2}. \tag{15.120}$$

Exercise 15.5. *Verify the expression for the optimal b_is in* (15.120).

15.5.2 Nonlinear Regression

In the last two subsections we have considered the case of linear regression where the parameters are linear functions. However, in this subsection we consider the case where we are fitting a model of the form

$$Y_i = f(\boldsymbol{x}_i, \boldsymbol{\beta}) + \boldsymbol{\varepsilon}_i, \tag{15.121}$$

where $f(\boldsymbol{x}_i, \boldsymbol{\beta})$ is a nonlinear function that relates the expectation of Y to the independent variables $\boldsymbol{x}_i$s. Examples of these nonlinear models are the exponential growth model, which is defined as

$$Y_i = \beta_1 e^{\beta_2 x_i} + \varepsilon_i, \tag{15.122}$$

or the Weibull model,

$$Y_i = \beta_1 e^{-\left(\frac{x_i}{\beta_2}\right)^{\beta_3}} + \varepsilon_i. \tag{15.123}$$

In general we seek, as for the linear regression cases, the least squares estimate of $\boldsymbol{\beta}$, denote by $\boldsymbol{b}$, that are the set of parameters that minimize the sum of squared residuals

$$R(\boldsymbol{\beta}) = \sum_{i=1}^{N} (Y_i - f(\boldsymbol{x}_i, \boldsymbol{b}))^2. \tag{15.124}$$

As with the linear regression cases, we can obtain the set of normal equations for this model in general as

$$\frac{\partial R(\boldsymbol{b})}{\partial \boldsymbol{b}_j} = -2 \sum_{i=1}^{N} (Y_i - f(\boldsymbol{x}_i, \boldsymbol{b})) \left(\frac{f(\boldsymbol{x}_i, \boldsymbol{b})}{\partial \boldsymbol{b}_j} \right). \tag{15.125}$$

However, it is often the case that the normal equations for the parameters are themselves nonlinear, and therefore an explicit solution for $\boldsymbol{b}$ may be difficult to obtain.

To obtain approximations for the parameters, it is often the case that an iterative solver is required. We shall go into detail about some of these solvers in Chapter 17.

15.6 Optimal (Optimum) Interpolation/Statistical Interpolation/Analysis Correction

Given the statistical and probability theory from the last section, we now move on to the last of the set of non-variational-based data assimilation schemes. The reason there are three names in the title of this section is that optimal interpolation, which was originally derived in the Soviet Union by Gandin in 1963, translated into English in 1965 [149], is often not considered as actually being an optimal interpolation. In fact the actual name for this method is **optimum interpolation**.

Statistical interpolation is the phase coined by Dr. Roger Daley in his very good book "*Atmospheric Data Analysis*," [85] but also in his publication [83]. The final name, analysis correction (AC) comes from a permutation of optimum interpolation theory that lead to the operational implementation at

the United Kingdom's Meteorological Office [258]. In this section we shall present all three different formulations, but we shall state that it is quite difficult to obtain a copy of Gandin's book. However, there is a good derivation along with an explanation of the practical aspect of the optimum interpolation method in a paper by Alaka and Elvander in 1972 [3]. It is the derivation from [3] that we summarize here.

15.6.1 Derivation of the Optimum Interpolation From Alaka and Elvander [3]

Let $\mathbf{r}_i = \mathbf{r}_0, \mathbf{r}_1, \ldots, \mathbf{r}_n$ denote a set of independent vectors defining the location of points in a sampling space. Next we consider a function $f(\mathbf{r})$ whose sampled values $\hat{f}_i = \hat{f}_1, \hat{f}_2, \ldots, \hat{f}_n$ have errors $\varepsilon_i = \varepsilon_1, \varepsilon_2 \ldots, \varepsilon_n$, so that

$$\hat{f}_i = f_i + \varepsilon_i. \tag{15.126}$$

The problem then is to determine the values of f_0 at some location $\mathbf{r}_0$ from the measured values $\hat{f}_i$. We now let f_0' and f_i' denote the deviations of f_0 and f_i from their respective mean states, which allows us to express f_0' in terms of the following linear combination:

$$f_0' = \sum_{i=1}^{n} \left(f_i' + \varepsilon_i\right) P_i + I_0, \tag{15.127}$$

in which P_i are the weighting factors and I_0 is the errors in determining f_0' by interpolating from $\hat{f}_i$.

We now define the mean square interpolating error as being given by

$$\epsilon = \overline{I_0^2} = \overline{\left(\sum_{i=1}^{n} \left(f_i' + \varepsilon_i\right) P_i - f_0'\right)^2}. \tag{15.128}$$

We now make the standard assumption that the random (background) errors, ε_i, are independent of the true values of the measured quantities, which means that $\overline{\varepsilon_i f_i'} = 0$, and that they are unrelated to each other, which implies

$$\overline{\varepsilon_i \varepsilon_j} = \begin{cases} 0 & i \neq j, \\ \sigma_{\varepsilon_i}^2 & i = j, \end{cases} \tag{15.129}$$

where in [3] $\sigma_{\varepsilon_i}^2$ is the mean-square random observation errors. The assumptions above imply that the random errors do not affect the values of the true covariances, but inflate the true variances σ_i^2 by an amount $\sigma_{\varepsilon_i}^2$.

Given the assumptions above, we can rewrite (15.128) as

$$\epsilon = \sum_{i=1}^{n} \sum_{j=1}^{n} \overline{\hat{f}_i' \hat{f}_j'} P_i P_j + \sum_{i=1}^{n} \sigma_{\varepsilon_i}^2 P_i^2 - 2 \sum_{i=1}^{n} \overline{\hat{f}_i' f_0'} P_i + \sigma_0^2. \tag{15.130}$$

The **optimum** weights, p_i, corresponding to a minimum value of ϵ, are obtained by setting

$$\frac{\partial \epsilon}{\partial P_i} = 0. \tag{15.131}$$

As we saw in the linear regression section, (15.131) forms the set of linear equations

$$\sum_{j=1}^{n} \overline{\hat{f}_i' \hat{f}_j'} p_j + \sigma_{\varepsilon_i}^2 p_i = \overline{\hat{f}_i' f_o'}, \quad i = 1, 2, \ldots, n. \tag{15.132}$$

If we now denote the minimum of the mean-square error, ϵ_{min}, by E, then combining (15.130) and (15.132), we obtain

$$E = \sigma_0^2 - \sum_{i=1}^{n} \overline{\hat{f}_i' f_0'} p_i. \tag{15.133}$$

Before we progress further with the derivation from [3], we introduce two new terms that play an important part in the many versions of data assimilation schemes.

Definition 15.6 (Homogeneity of Variances). Homogeneity of variances assumes that the dependent variables exhibit equal levels of variance across the range of predictor variables. This is also true for the covariances and is referred to as the **homogeneity of covariances**.

Definition 15.7 (Isotropic). A covariance is said to be **isotropic** if it is uniform in all directions. This implies that the values are the same even under a rotation of the points.

We now assume the two properties defined above for the variances (homogeneous) and for the covariances (homogeneous and isotropic). This enables us to write (15.132) as

$$\sum_{j=1}^{n} \mu_{i,j} pj + \lambda_i^2 p_i = \mu_{0,i}, \quad \text{for } i = 1, 2, \ldots, n, \tag{15.134}$$

and (15.133) as

$$E = \sigma^2 \left(1 - \sum_{i=1}^{n} \mu_{0,i} p_i \right), \tag{15.135}$$

where

$$\mu_{i,j} = \frac{\overline{\hat{f}_i' f_j'}}{\sigma^2},$$

is the autocorrelation coefficient between values of the function at locations $\mathbf{r}_i$ and $\mathbf{r}_j$, while

$$\mu_{0,j} = \frac{\overline{\hat{f}_0' f_i'}}{\sigma^2},$$

is the autocorrelation coefficient between values of the function at $\mathbf{r}_0$ and $\mathbf{r}_i$, and finally

$$\lambda_i^2 = \frac{\sigma_{\varepsilon_i}^2}{\sigma^2}.$$

A remark that is now made in [3] is that from Eqs. (15.134) and (15.135), we see that both the weights and the root mean square interpolation errors depend on the scale of the function as represented by the autocorrelations $\mu_{0,i}$, $\mu_{i,j}$, on the variability as represented by σ_2, and on the random error σ_{ε_i}.

Another important feature to note here is that (15.135) informs us that the mean square interpolation error cannot exceed the variance of the function that is being interpolated.

For us to be able to effectively minimize the root mean square interpolation error through the procedure just described, we require accurate estimates of the random errors, the variance, and the autocorrelations functions. We have assumed that the variances are homogeneous, and that the covariances are homogeneous and isotropic. Therefore, to determine the root mean square random error, σ_{ε_i}, we assume a structure function, β, to be homogeneous and isotropic. Under these assumptions, this function depends only on $\boldsymbol{\rho} = \mathbf{r}_i - \mathbf{r}_j$, which is the distance between observation pairs located at $\mathbf{r}_i$ and $\mathbf{r}_j$. Therefore,

$$\beta(\boldsymbol{\rho}) = \overline{\left(f_i' - f_j'\right)^2}. \tag{15.136}$$

Now it is stated in Alaka and Elvander that the estimated structure function, $\hat{\beta}(\boldsymbol{\rho})$, is related to the true function $\beta(\boldsymbol{\rho})$ through

$$\hat{\beta}(\boldsymbol{\rho}) = \beta(\boldsymbol{\rho}) + 2\sigma_\varepsilon^2, \tag{15.137}$$

but they reference Chapter 2 from Gandin's book as their source for (15.137).

Alaka and Elvander then state a procedure to obtain estimates for $2\sigma_\varepsilon$ term as that of fitting a curve to the computed structure function $\beta(\boldsymbol{\rho})$ plotted against distance $\boldsymbol{\rho}$ and extrapolating the curve until it intersect the axis of $\beta(\boldsymbol{\rho})$ at $\boldsymbol{\rho} = \mathbf{0}$. It is then stated that the value of σ_ε^2 will comprise of both the random measurement errors and the aliasing errors inherent in the observations.

Determining the variances

In the experiment that Alaka and Elvander consider in [3], they are using 10 years of rawinsonde observations from 35 stations in the Caribbean region, and they only used station pairs with at least 100 simultaneous observations. They state that it is well known that variances computed from data having random errors are too high by an amount equal to the mean square values of these errors. They then state that in their study they calculate the variances at each station and then these are corrected through the subtraction of $\sigma\varepsilon^2$, which has been calculated through the procedure described above. Finally, in [3] they apply and average over the stations to ensure homogeneity of the variance.

Determining the autocorrelation functions

Under the assumption of homogeneity and isotropy for the autocorrelations, μ is function of the distance, $\boldsymbol{\rho}$, between observation pairs, and so can be written as $\mu(\boldsymbol{\rho}) = \frac{\overline{f_i' f_j'}}{\sigma^2}$.

However, in [3] the authors state that when they plotted the autocorrelations, the results contained scatter which was a result of the anisotropy and non-homogeneity of the true autocorrelations. To smooth these non-conforming autocorrelations, the points were divided into 100 km segments, with the middle of these intervals stored in the vector $\boldsymbol{d}$, and then given the number of points in each interval N_i, they use the following function:

$$\mu(\boldsymbol{d}) = \sum_{i=1}^{n} \hat{\mu}(\rho_i) \frac{N_i\left(\frac{1}{2} + \frac{1}{2}\cos\frac{\pi(\rho_i - d)}{100}\right)}{\sum_{j=1}^{n} N_j\left(\frac{1}{2} + \frac{1}{2}\cos\frac{\pi(\rho_i - d)}{100}\right)}, \tag{15.138}$$

where ρ_i denoted the distance between the pairs of stations used in the calculations.

The values $\mu(\boldsymbol{d})$ obtained through (15.138) were then fitted to the empirical curve of the form

$$\mu(\boldsymbol{\rho}) = \left(Ae^{-B\rho^c} + 1 - A\right)\cos D\rho. \tag{15.139}$$

We have recreated the four plots of the autocorrelations of the zonal winds from [3] at 850 and 200 mb in January and July in Fig. 15.4. We can see that they detected seasonal changes as well as changes with respect to height. The determining of autocorrelation and variances is a task that still affects the performance of the data assimilation schemes today.

Determining the optimum weights

The autocorrelation functions play an important role in determining the relative weights to be given to observations at different distances from the location to which the interpolation is made. [3] presents results from an experiment where they show a schematic of a 12-point interpolation grid that comprises of the central point, which is where we are interpolating to, and an inner ring of four observations, and finally an outer ring that comprises of eight observations. A replica of this grid is presented in Fig. 15.5.

As we have seen from the summary of [3], there are many different aspects involved in determining the different parts of the optimum interpolation. We now consider the matrix version of the expression for the optimum interpolation scheme.

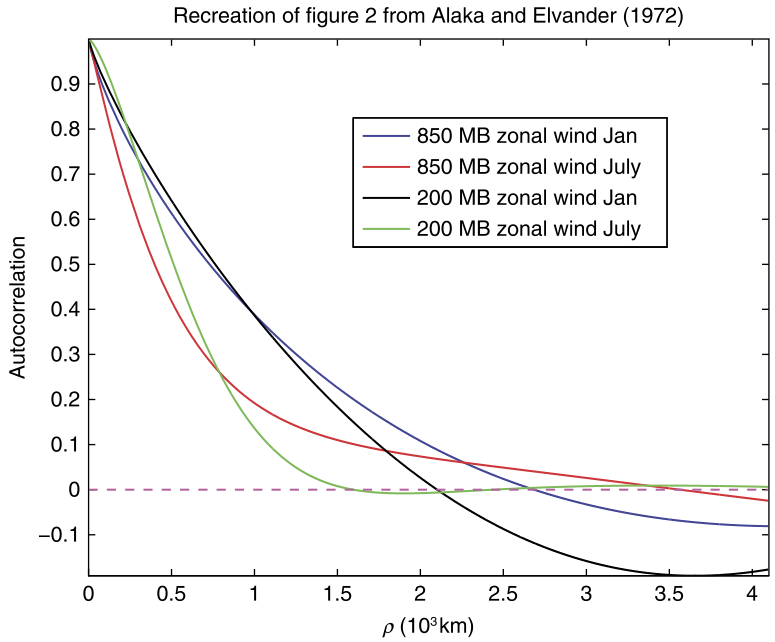


FIGURE 15.4

Recreation of the autocorrelation plots from Alaka and Elvander (1972) for the zonal winds at 850 and 200 mb in January and July.

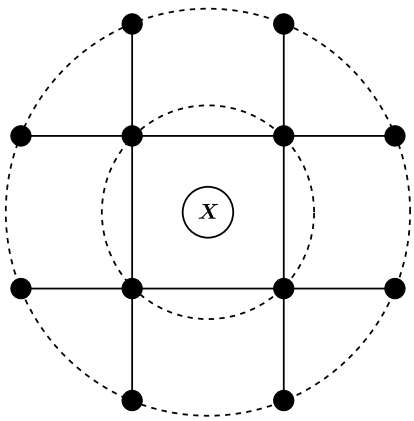


FIGURE 15.5

Recreation of the 12-point interpolation grid used in Alaka and Elvander [3].

15.6.2 Matrix Version of Optimum Interpolation

The matrix form of the optimum interpolation equation has already been derived in the multivariate least squares section, which is

$$\boldsymbol{x}_a = \boldsymbol{x}_b + \mathbf{W}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}\right)\right), \tag{15.140}$$

where the optimal weight matrix that minimizes the analysis error covariance matrix is given by

$$\mathbf{W} = \mathbf{B}\mathbf{H}^T\left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R}\right)^{-1}. \tag{15.141}$$

If the background error covariances and the observational error covariances are known exactly, then (15.140) and (15.141) provide the **optimal interpolation**. Therefore, the analysis error covariance matrix is given by

$$\mathbf{P}_a = \left(\mathbf{I} - \mathbf{B}\mathbf{H}^T\left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{H}\right)\mathbf{B}. \tag{15.142}$$

However, in reality we do not know the statistics involved exactly and only have an approximation to the true statistics. This is the reason why this scheme is referred to as optimum, or as **statistical interpolation**, which is how Daley refers to the scheme in [83,85].

15.6.3 Implementation of OI

As we saw in the derivation from Alaka and Elvander when they implemented their version of OI, they had specific autocorrelation functions, but also a set distance for which observations for that grid point would be interpolated to, weighted by their distance. This raises two points about the implementation of the OI scheme. The first point is the type of autocorrelation function that is used, which is a function of distance, so a subpoint is how far out do the correlations extend, and the second point is about the

grid, or as some centers used, volumes, that will contain the observations that will be weighted to each grid point.

We shall take each point in turn and provide brief summaries of different applications in different geophysical fields.

Atkins [16]

In [16] the author studies the objective analysis of relative humidity, but here she relaxes the isotropy assumption of the weighting functions. It is stated in the abstract that the gradient of the background field is used in such a way that the observations in a direction along closely packed isopleths have higher weight than those in a direction perpendicular to them. The distance weighting function used in [16] was

$$p_i = \frac{1}{1 + 0.01 r_i^n}, \tag{15.143}$$

where r_i is the distance from the grid point to the observation in grid length. The scheme presented in [16] appears to have a mixture of OI and the Cressman scheme, in that it performs two scans, where the second scan is over a much smaller region. We have plotted the weighting factors from [16] in Fig. 15.6.

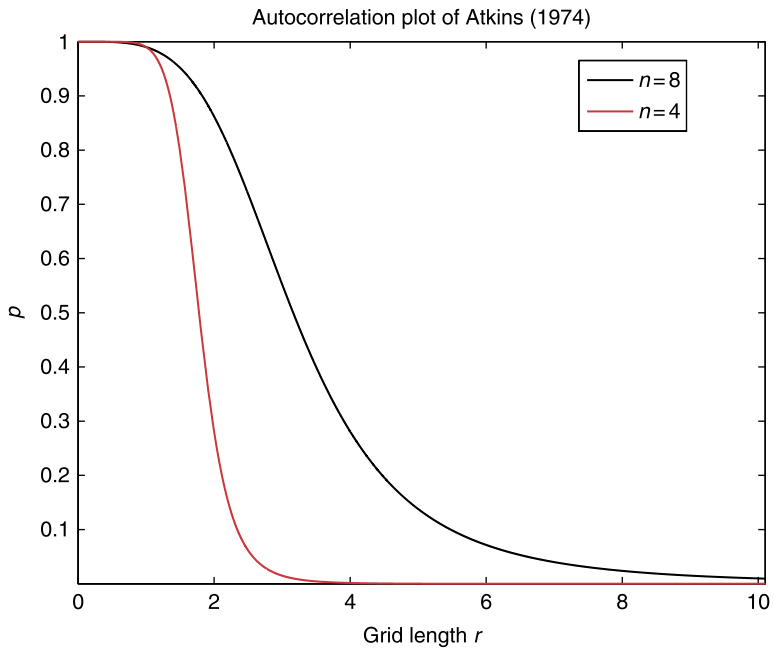


FIGURE 15.6

Recreation of the autocorrelation plot from Atkins (1974), [16] which on the first sweep $n = 8$ and on the second sweep $n = 4$, which we can see drastically reduces the lengths of the correlations.

Bergman [30]

One of the desired properties of a statistical interpolation scheme is to ensure that the balances that are present, either in the atmosphere, or in the oceans, are still maintained. There was a lot of work in the 1970s and 1980s to find ways to impose balance into the analysis scheme. The scheme developed in [30] focused on updating temperature and horizontal wind components at the grid points. The multivariate aspect of the approach in [30] is to use wind observations in the interpolative analysis of the temperature field and vice versa in the form of the thermal wind relationship. This approach had been suggested earlier in [101]. The reason for this approach is to try to obtain a form of mass-momentum balancing provided by the analysis, which Bergman says would not be present if the two fields were analyzed separately. A full derivation of the correlation between different aspects of the temperature and wind fields can be found in [30]. In Fig. 15.7 is a copy of figure 1 from [30] to help illustrate the type of structures that this approach is able to create in the correlation functions.

The manner of which the covariance model was designed in [30] was mapped to a stereographic projection, where given a point at i, the x and y coordinates in this projection were given by

$$xi = am_i \cos\theta \cos\lambda, \tag{15.144a}$$

$$y_i = am_i \cos\theta \sin\lambda, \tag{15.144b}$$

where a is the radius of the Earth, and the m_is are the map factors that are true at the North Pole. The northern and southern hemispheres' mapping factors are

$$m_i = \frac{2}{1+\sin\theta}, \quad m_i = \frac{2}{1+\sin -\theta}. \tag{15.145}$$

The distance between two points is given by

$$\Delta s_{ij} = \frac{2}{m_i + m_j}\sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2}. \tag{15.146}$$

The covariance model that is used in [30] is based upon [437] and is a form of an exponential given by

$$\mu_{ij}^{tt} = e^{-k_h\left(\Delta s_{ij}\right)^2}. \tag{15.147}$$

Thiebaux [437]

A spectral-based geostrophic-based balanced autocorrelations was suggested in [437] as a means of determining the covariance matrix for the optimum analysis scheme Gandin presented in 1963 [149]. His motivation was to make sure that the choice of autocorrelation function for geopotential ensured that there were geostrophically consistent cross-correlations, and therefore, by association, covariances for height and winds. There are four different correlations functions that Thiebaux investigated, but there is an aside in [437] that is very insightful about why we have to be careful about the choice of covariance model:

> To the extent that the atmosphere is consistently anisotropic, a function solely of distance between points must be an inaccurate representation of autocorrelation. And, insofar as the approximate geostrophic relations

$$u = -k\frac{\partial h}{\partial \theta}, \text{ and } v = k\frac{1}{\cos\theta}\frac{\partial h}{\partial \lambda}$$

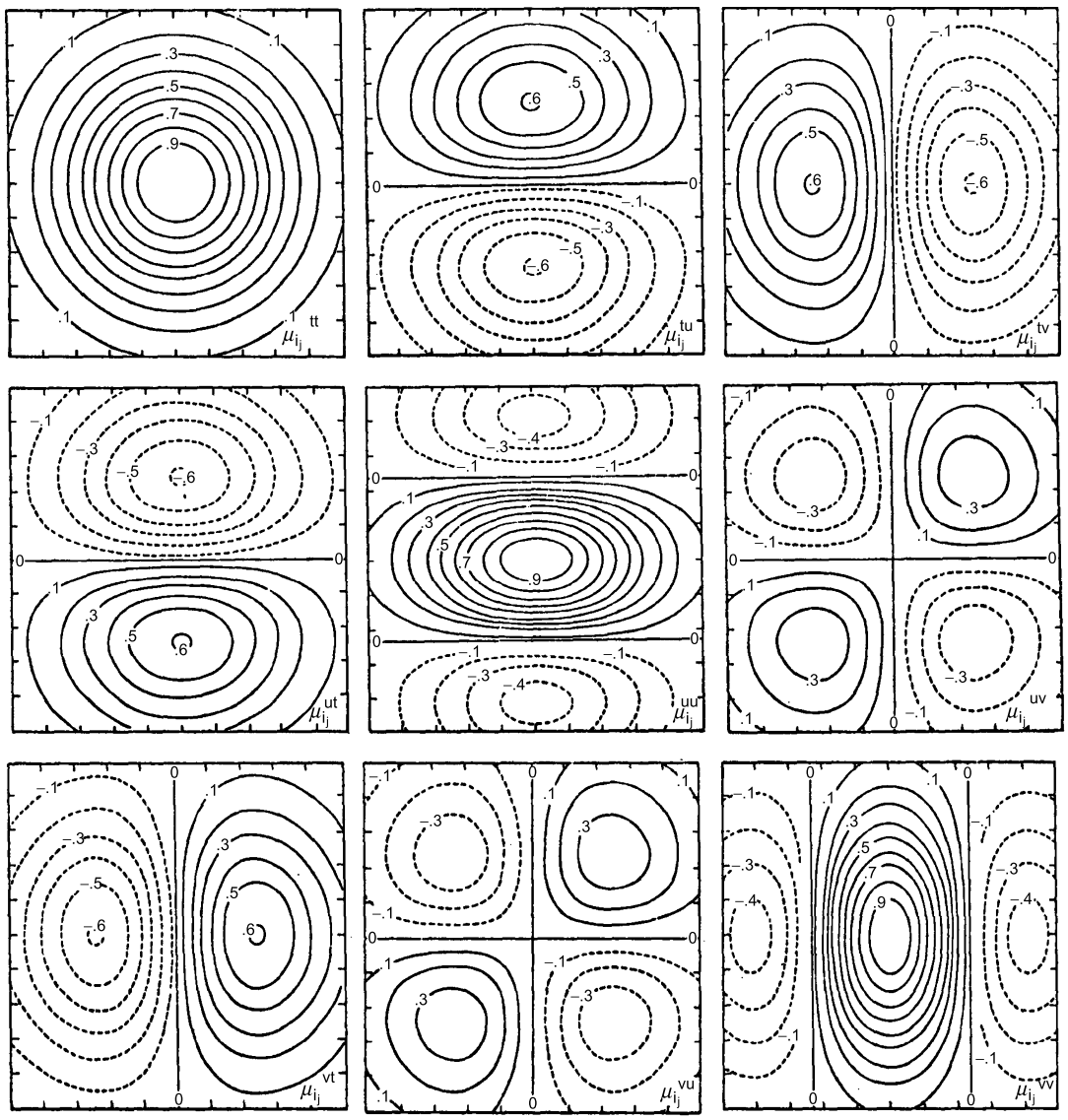


FIGURE 15.7

Isopath of lateral correlation functions. Figure 1 from Bergman, K. H. (1979). Multivariate Analysis of Temperatures and Winds Using Optimum Interpolation, Monthly Weather Review, 107(11), 1423-1444. **© American Meteorological society**.

fail to represent true atmospheric dynamics, derivations of cross correlations from them incorporates further loss of accuracy. Since the accuracy of the objective analysis depends upon the accuracy of representation of covariance structure within the analysis region, each failure of modeled statistics to represent true statistics may be expected to degrade the quality of the analysis.

Lorenc [258]

A global three-dimensional multivariate OI scheme is introduced in [258] which would be the basis of the OI scheme at ECMWF throughout the 1980s and the early part of the 1990s. The motivation in [258] was to develop a scheme to provide initial states for ECMWF's operational forecast model. The scheme needed to produce global analyses in numerical form, efficiently, and here is the important part which makes the scheme different: "with minimal human intervention" on a large fast vector-processing computer.

The OI-based scheme described in [258] is credited in being the first fully three-dimensional version of optimum interpolation. It is three dimensional because it has multivariate relationships between the wind, height, and thickness. The covariance matrix that Lorenc introduces in [258] is based on the separability of the vertical component from the horizontal. We recommend reading [258] to learn about the development toward the multivariate version of OI. Lorenc defines a series of covariances that link the fields through different balance assumptions.

The motivation for the formulation of the covariance model for a statistical interpolation scheme in [83] is primarily concerned with aiding in the analysis of the divergent component of the wind. As in [258], Daley in [83] defines a set of covariances between the different field, but now with a divergent wind covariance model in the horizontal. The distances between the arbitrary points, i and j, are defined by

$$r_{ij} = \left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2, \tag{15.148}$$

where, given these distances the covariance model in the horizontal is the Gaussian model, which is given by

$$F\left(r_{ij}^2\right) = e^{-\frac{r_{ij}}{2}}. \tag{15.149}$$

Barker [23]

In [23] we see the development of an operational configuration of a multivariate OI scheme, where it is assumed that the correlation model can be expressed as the product of the horizontal and vertical covariances in the form

$$\langle p_1 p_2 \rangle = \left\langle p^2 \right\rangle H\left(r\right) V\left(h_1, h_2\right),$$

where $H\left(r\right)$ is the horizontal structure function, r is the horizontal distance between observation locations 1 and 2, $V\left(h_1, h_2\right)$ is the vertical structure function, and h_1 and h_2 are the heights corresponding to the pressure level of observation locations 1 and 2.

The interesting feature about the covariance model used in what was referred to the Navy's Multivariate Optimum Interpolation (MVOI) was that the blending weights incorporated the relationships

among geophysical height, winds, and geopotential thickness between pressure surfaces. The correlation models were initially defined in terms of geopotential ϕ, stream function, ψ, and velocity potential, χ, as

$$\begin{aligned}\langle \phi_i \phi_j \rangle &= \left\langle \phi^2 \right\rangle E(r), \\ \langle \phi_i \phi_j \rangle &= \left\langle \psi^2 \right\rangle F(r), \\ \langle \chi_i \chi_j \rangle &= \left\langle \chi^2 \right\rangle G(r),\end{aligned}$$

where $\langle \phi^2 \rangle$, $\langle \psi^2 \rangle$, and $\langle \chi^2 \rangle$ are the error variances of the predicted estimates, and $E(r)$, $F(r)$, and $G(r)$ are the horizontal correlation models given by

$$\begin{aligned}E(r) &= 1 - a + a(1 - br) R^{-br}, \\ F(r) &= (1 - \nu) E(r), \\ G(r) &= \nu E(r),\end{aligned}$$

where ν is the divergence parameter that relates the covariance models for stream function F and velocity potential G to the model for the geopotential height, and a and b are empirical parameters to be set, in [23] the values of $a = 0.9$ and $b = 2.6$ are used. There are other couplings that are enforced through geostrophy and the divergence. The vertical covariance model is given by

$$V(h_1, h_2) = \exp\left\{ -\left(\frac{|h_1 - h_2|}{H} \right)^{1.8} \right\},$$

where $H = 3600$ m.

At the same time that the United States Navy was developing their optimum interpolation scheme for the atmosphere, another group of scientists at the Fleet Numerical Oceanography Center in Monterey, California, along with scientists at the Stennis Space Center in Mississippi, were developing an optimum interpolation scheme for their operational global scale ocean thermal analysis system [63].

15.6.4 Analysis Correction (AC)

The AC scheme was introduced as the United Kingdom's Meteorological Office's operational data assimilation system in 1991. This scheme replaced the optimum interpolation operational system due to the costly calculations of the selection and weighting of data. The AC, hereafter, has similarities between the successive methods of [32,79].

The starting point for the AC method is to assume that the probability density functions of observational error, and in the errors of the observation operator $\boldsymbol{h}$, are approximately Gaussian, with covariance matrices $\mathbf{O}$ and $\mathbf{F}$, respectively, this implies that the fit to observations is measured by

$$(\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}))^T (\mathbf{O} + \mathbf{F})^{-1} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x})). \tag{15.150}$$

The analysis problem can be thought of as finding the generalized inverse of $(\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}))$ to interpolate from observations to model.

When there are no observations present in a specific region, the best estimate is taken of the most likely model state which is referred to as the **background**, and is denoted by $\boldsymbol{x}_b$. If the errors in the

prior knowledge are sufficiently linear and Gaussian, they can be expressed by the background error covariance matrix $\mathbf{B}$, and deviations from the prior knowledge can be measured by

$$(\boldsymbol{x}_b - \boldsymbol{x})^T \mathbf{B}^{-1} (\boldsymbol{x}_b - \boldsymbol{x}) . \tag{15.151}$$

The basic optimal analysis problem is therefore to simply minimize a penalty functional (J) given by the sum of (15.150) and (15.150) which is

$$J = (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}))^T (\mathbf{O} + \mathbf{F})^{-1} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x})) + (\boldsymbol{x}_b - \boldsymbol{x})^T \mathbf{B}^{-1} (\boldsymbol{x}_b - \boldsymbol{x}) . \tag{15.152}$$

An important feature to note about (15.152) is that it is very similar to what we shall introduce as 3D VAR in the next chapter, but here we carry on following the derivation of the AC approach from [263].

The next step in the derivation of the AC theory is to consider iterative solutions to the minimum of the functional in (15.152), which is achieved through finding the zeros of the derivative of (15.152) with respect to $\boldsymbol{x}$. This then yields

$$\boldsymbol{J}' = -2 \left(\mathbf{H}^T (\mathbf{O} + \mathbf{F})^{-1} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x})) + \mathbf{B}^{-1} (\boldsymbol{x}_b - \boldsymbol{x}) \right) , \tag{15.153}$$

where $\boldsymbol{J}'$ is the vector of the partial derivatives of J and is referred to as the **Jacobian**.

Now we have to derive the second derivative of J, which is referred to as the **Hessian**, and is a **matrix**, defined by

$$\mathbf{J}'' = 2 \left(\mathbf{H}^T (\mathbf{O} + \mathbf{F})^{-1} \mathbf{H} + \mathbf{B}^{-1} \right) . \tag{15.154}$$

An important comment is made about (15.154) in [263] in that the vectorial component of the Hessian is not evaluated. Therefore, if we were to apply a Gauss–Newton (spoiler) iteration, we have

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \left(\mathbf{J}'' \right)^{-1} \boldsymbol{J}' . \tag{15.155}$$

We now introduce the component that resembles the successive correction schemes from [24,32,79], where a factor $\mathbf{B}^{-1}$ has been moved from the Jacobian to the Hessian of the functional so that

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \mathbf{Q}\mathbf{W} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_n)) + \boldsymbol{x}_b - \boldsymbol{x}_n , \tag{15.156a}$$

$$\mathbf{W} = \mathbf{B}\mathbf{H}^T (\mathbf{O} + \mathbf{F})^{-1} , \tag{15.156b}$$

$$\mathbf{Q} = (\mathbf{W}\mathbf{H} + I)^{-1} , \tag{15.156c}$$

where W is an $N \times N_o$ matrix of weights, and $\mathbf{Q}$ is an $N \times N$ matrix of normalization, where N is the number of model variables and N_o is the number of observations.

There are some approximations made in [263] that enables us to remove the last two terms in (15.156a), which results in the successive correction equation as

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \mathbf{Q}\mathbf{W} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_n)) , \tag{15.157}$$

where the weights w that form the matrix $\mathbf{W}$ are given by a formula that is similar to (15.156b), and the normalization factors $\mathbf{Q}$ are given by a diagonal matrix whose entries are $\left(\sum_{i=1}^{N} w_i + 1 \right)^{-1}$.

An important remark made in [263] at the end of the subsection on successive correction is that by removing the forcing toward the background, the method will converge to the observations exactly, if this is possible. This implies that the iteration in (15.157) will only converge to the optimal analysis in the case where the observations are perfect, i.e., no observational error.

The reason why this scheme is referred to as the AC, rather than as a successive correction, is because it attempts to compensate for the observation density, whereas the successive correction scheme gives each observation in date-dense areas equal weights to isolated observations.

The last modification that is made to the successive correction approach in the AC scheme is to make the weights dependent on the local observation density, which is achieved by replacing the grid point normalization $\mathbf{Q}$, which has already been approximated by a diagonal matrix, by a similar observation normalization $\tilde{\mathbf{Q}}$ evaluated at observation positions which is also a diagonal matrix. Given all these assumptions, the modified successive correction scheme is given by

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \mathbf{W}\tilde{\mathbf{Q}}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_n\right)\right), \tag{15.158}$$

which is the form that the AC scheme at the Met. Office, as it is now referred, took.

There are a lot more details about the implementation of the AC scheme in [263], including the horizontal second order autoregressive correlation function that is a function of distance r_{ij} from observation i to grid point j and the correlation scale s, which is defined as

$$\mu_{ij} = \left(1 + \frac{r_{ij}}{s}\right) e^{-\frac{r_{ij}}{s}}.$$

There is still much debate about which autoregressive and autocorrelation lengthscales to use in many forms of data assimilation.

15.7 Summary

In this chapter we have introduced the earlier forms of data assimilation that were used in both atmospheric and ocean problems. There were two types of methods that we introduced: the empirical methods and nudging, and then the three different forms of successive corrections. These types of schemes are referred to as objective analysis, although we must note that this term also applies to the optimum interpolation-based schemes, which were the second form of data assimilation, but where these algorithms were statistically based. We have introduced the least squares principle in univariate and multivariate formulation in the linear and nonlinear. For the nonlinear least squares approach, we summarized results from the paper by Tarantola and Valette in 1982, which appears to have the equation for the nonlinear iterative solution to incremental 3D VAR.

Finally we have introduced the optimum interpolation schemes, often mistakenly called the optimal interpolation schemes, which they are not because we do not know the correct form for the error covariance matrices that are involved. We have introduced different functions that different researchers have used over the years to ensure that dynamical balances are maintained in the solutions to the OI schemes so that the numerical model will not excite spurious waves. We finished this chapter with the introduction of the AC scheme which was the United Kingdom's Meteorological Office's operation system in the early 1990s until it was replaced by the incremental 3D VAR scheme.

The reason for this chapter is not as a history lesson in the evolution of data assimilation methods, but to demonstrate to readers that this theory still works, and is still in use today. For example the NOAA sea surface temperature analysis [365] uses OI. OI has also been used recently in the analysis of Aquarius satellite observations between 2011 and 2015 of sea surface salinity [292]. Staying with oceans, OI has also recently been used for a quality assessment of high-frequency radar derived surface currents [116], where the authors are carrying out a study to see the impact of the radars being outside the optimal region for the placement of the radars to scan the sea surface. They compare the performance of the OI to an unweighed least squares fit and they show that the OI scheme can reproduce quite good estimates under certain circumstances.

The methods that we have introduced in this chapter, as we just said, are still in use today and are being evolved to adapt to the constraints of the different geophysical systems that we are interested in. We now move on to the variational, optimal control, probabilistic-based methods for data assimilation, starting with the **variational methods**, or VAR as it is more often called.

This page intentionally left blank

CHAPTER

Variational Data Assimilation 16

Contents

Variational-based methods for data assimilation have been in operational use for ocean and weather prediction for quite some time. National Center for Environmental Prediction (NCEP) was the first to go operational with 3D VAR, followed not too far behind by the European Center for Medium-range Weather Forecasting (ECMWF) and the United Kingdom's Meteorological Office, along with most of the operational centers for global numerical weather prediction. The reason for the introduction of 3D VAR was so that data assimilation schemes could globally model three-dimensional covariances for the errors of the background field, but also use nonlinear observation operators, especially those associated with satellites. This was a problem for the optimum interpolation-based methods, as they are quite reliant on the observation operator being linear.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00020-6

Three-dimensional, or non-temporal, variational assimilation systems are calculus of variations based data assimilation schemes that do not include the time component in the assimilation systems. The name for these types of systems are 1D VAR and 2D VAR. However, there is some confusion about what to call these schemes when the time component is included. For clarity here, we shall refer to **any** dimensional scheme that has the time component as a 4D VAR system.

The introduction of the time component in the analysis scheme enables the assimilation of observations through a **window**, which is the length of time between analysis times. Schemes that do not include the time component are referred to as **filters**, while those that do contain the time components are **smoothers**. The reason for these names can be seen in the schematic shown in Fig. 16.1.

We can see that the filter-based methods (Fig. 16.1A) use observations that are available at the analysis time, and as such the data assimilation scheme is filtering the observations and the dynamical scales to find the best estimate of the geophysical state at that time to fit to the observations. The estimation at this time could be to find a new set of initial conditions at that time, but it could also be to update the model, and/or observational parameters, to better fit to the observations or to stay closer to the model. However, for the schemes with the time component, Fig. 16.1B, we seek the best trajectory that minimizes the distances between the observations and the trajectory through the whole window and as such we are smoothing the trajectory.

There is also another technique for the filtering approach in a variational context which is referred to as the **First Guess at Appropriate Time (FGAT)**; here observations that are within a set distance either side of the analysis time are *interpolated* to the analysis time to be assimilated as extra observations [268]. The reason for using a FGAT is to ensure that vital observational information of the system is not lost due to operational restrictions on the time to complete the analysis cycle.

Given this overview, we now move on to the mathematical, statistical, and probabilistic theory that enables us to have these variational approaches.

16.1 Sasaki and the Strong and Weak Constraints

In the previous chapter we introduced the different statistical-based schemes that were used for data assimilation before the introduction of the variational-based schemes. The emphasis here is on the statistical component. All of the various schemes, successive corrections, Barnes scheme, optimum interpolation, and analysis correction were based on minimizing a form of sum of squares, which for a linear Gaussian case is equivalent to finding the state that has the minimum variance. At the same time as the development of these scheme, there was an approach based upon calculus of variations being developed by **Yoshikazu Sasaki**.

Sasaki wrote a series of papers advocating that the variational principle could be used to form an objective analysis for numerical weather prediction [383–388]. The two fundamental papers associated with Sasaki's work are: *An Objective Analysis Based Upon Variational Methods* [384] and *Some Basic Formalisms in Numerical Variational Analysis* [386]; it is the latter paper that we shall briefly summarize here.

The starting point for the calculus of variational approach, which we shall refer to as just variational from here on, is to define the functional, J, as

$$\delta J = \delta \sum_{\Omega} \sum_{i} \left(\tilde{\alpha}_i \left(\varphi_i - \tilde{\varphi} \right)^2 + \alpha_i \left(\nabla_t \varphi_i \right)^2 \right) = 0, \tag{16.1}$$

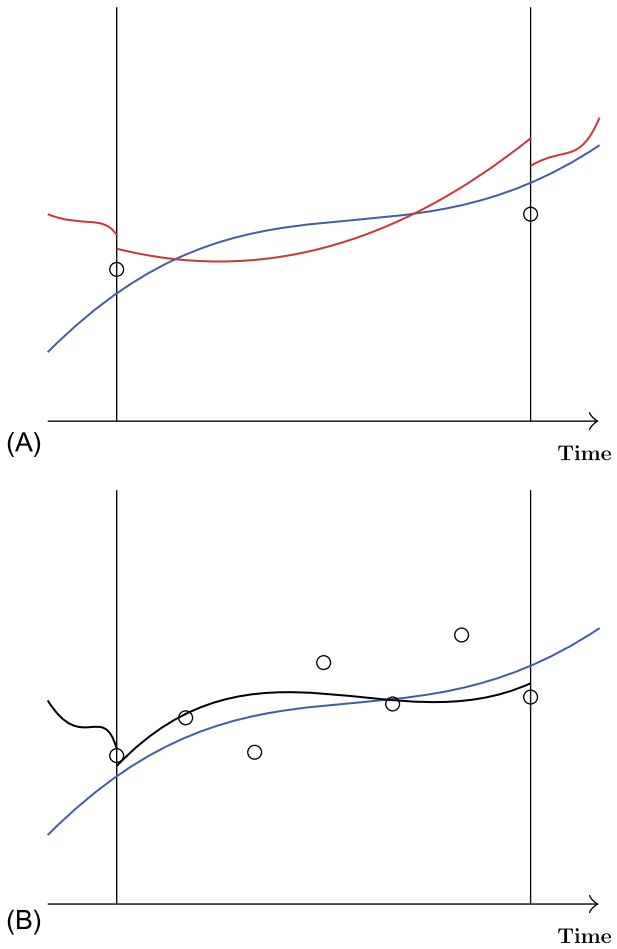


FIGURE 16.1

(A) Schematic of filtering, where the blue line is the true trajectory, the red line is the 3D VAR trajectory, and the circles are the observations at the analysis times. (B) Schematic of smoothing where the blue line is the true trajectory, the black line is the 4D VAR smoothed trajectory, and the circles are observations throughout the window.

where δ is the variational operator, φ_i is the analyzed field, $\tilde{\varphi}_i$ is the observation, ∇_t is the local change in a finite-difference form, $\tilde{\alpha}_i$ and α_i are predetermined weights, and Ω is the domain in time t and space x_1, x_2, x_3. Sasaki refers to the first term in (16.1) as a condition used for minimizing the variance of the difference between the observed and analyzed values. The second term is a simple low pass filter in frequency. (16.1) is solved with the dynamical constraint such as those given by the primitive equations. It is possible to write these constraints in the form

$$\nabla_t \varphi_i = F_i \left(\varphi_i, \varphi_j, \nabla_{x_k} \varphi_i, \nabla_{x_k} \varphi_j \right), \tag{16.2}$$

where F_i is a given function and ∇_{xk} represents the space derivative with respect to x_k for $k = 1, 2, 3$.

The functional defined in (16.1) is quadratic and therefore the associated stationary value of J becomes the minimum. As we saw in the calculus of variation and optimal control chapters, Chapters 5 and 7, respectively, the solution of (16.1) is obtained through solving the Euler, or the Euler-Lagrange, equations after the substitution of (16.2) into (16.1). Sasaki states that the disadvantage of this approach is that it is only for an instantaneous field and the functional does not describe explicitly the **time variations**.

This disadvantage can be overcome by taking the following approaches. The first is an orthodox approach and is written as

$$\delta J = \delta \sum_{\Omega} \sum_{i} \left(\tilde{\alpha} \left(\varphi_i - \tilde{\varphi}_j \right)^2 + \lambda_i G_i \left(\varphi_i, \varphi_j \nabla_t \varphi_i, \nabla_{x_k} \varphi_i, \nabla_{x_k} \varphi_j \right) \right) = 0, \tag{16.3}$$

where G_i represents a prognostic or diagnostic equation and λ_i is the Lagrange multiplier. The Euler equation derived from (16.3) will include $\nabla_t \varphi_i$ and $\nabla_t \lambda_i$. It is noted here in [386] that the solution of the Euler equation requires a considerable amount of effort to solve numerically.

An alternative approach is to formulate the functional as

$$\delta J = \delta \sum_{\Omega} \sum_{i} \left(\tilde{\alpha}_i \left(\varphi_i - \tilde{\alpha}_i \right)^2 + \alpha_i G_i^2 \right) = 0, \tag{16.4}$$

where α_i is a predetermined weight. We should note that G_i is linear in (16.3) and quadratic in (16.4), and that the coefficient of the G term is the Lagrange multiplier in (16.3), but is a weight in (16.4). Because of these differences with the coefficients of G, we obtain the following two conditions:

$$G = 0, \tag{16.5a}$$

$$G \neq 0, \tag{16.5b}$$

from (16.3) and (16.4), respectively. Sasaki refers to formalism that results in (16.5a) as the **strong constraint**, and to the formalism that leads to (16.5b) as the **weak constraint**. We refer the reader to [386] for an example with the one-dimensional advection equation for the differences these two approaches form. However, we shall introduce loose definitions of what strong and weak constraint mean:

Definition 16.1. Strong constraint: This is the constraint where the analyzed solution to (16.3) must satisfy the discrete model exactly. The strong constraint is also stated as the **perfect model** assumption. This is also referred to as the case where there is **no model error**.

Definition 16.2. Weak constraint: This is the constraint where the analyzed solutions to (16.4) **does not** have to satisfy the discrete model equations exactly. The weak constraint is also stated as the **imperfect model** assumption. This is also referred to as the case where there **is model error**.

These constraints just described/derived apply to the four-dimensional (temporal-spatial) problems, but if we do not have the time component, and we wish to form an inverse problem to solve for a set of initial conditions for a numerical model, then it became apparent in the 1970s and early 1980s that a priori information was required to constrain the ill-posed problem [252]. This led to the beginning of the **Bayesian** approach for data assimilation. Therefore, we move on to 3D VAR and present the theory from [259].

16.2 Three-Dimensional Data Assimilation

From the 1986 paper by Andrew Lorenc [259], we have that the general starting point for the derivation of the 3D VAR cost function is to consider the problem of finding the set of initial states so that the subsequent forecast is the *best* possible. Due to the problem of the forecast being imperfect, we have to try to compensate by introducing observations. Therefore, let the state vector be $\boldsymbol{x}$, where $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)^T$, N is the total number of state variables, and $\boldsymbol{y}$ is the observational vector, where $\boldsymbol{y} = \left(y_1, y_2, \ldots, y_{N_o}\right)^T$, and N_o represents the number of observations, with $N_o \ll N$.

We now require a relationship between the model states, $\boldsymbol{x}$ and the observations, $\boldsymbol{y}$. This relationship is given by

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{16.6}$$

where $\boldsymbol{h}(\boldsymbol{x})$ is a vector of nonlinear interpolations from the model states to the observations given by

$$\boldsymbol{h}(\boldsymbol{x}) = \begin{pmatrix} h_1(x_1, x_2, \ldots, x_N) \\ h_2(x_1, x_2, \ldots, x_N) \\ \vdots \\ h_{N_o}(x_1, x_2, \ldots x_N) \end{pmatrix}. \tag{16.7}$$

If the relationship between the observation and the model state variables is linear, for example, an average or a linear interpolation to grid points, then $\mathbf{h}$ is a matrix vector multiplication, $\mathbf{H}\boldsymbol{x}$, where $\mathbf{H}$ is a linear rectangular matrix of dimensions $N_o \times N$. This then gives us the problem, according to Lorenc, of finding the "best" $\boldsymbol{x}$ that inverts (16.6) for a given $\boldsymbol{y}^o$, where $\boldsymbol{y}^o$ is the vector of physical observation that contains errors.

The method to set up this problem is to consider a Bayesian probability approach, where Bayes' theorem states that the posterior probability of an event A occurring, given that event B is known to have occurred, is proportional to the prior probability of A, multiplied by the probability of B occurring given that A is known to have occurred:

$$P(A \mid B) \propto P(B \mid A)\, P(A). \tag{16.8}$$

In the case that we are interested in A is the event that $\boldsymbol{x} = \boldsymbol{x}_t$ and B is the event $\boldsymbol{y} = \boldsymbol{y}^o$. This enables us to write (16.8) as

$$P\left(\boldsymbol{x} = \boldsymbol{x}_t \mid \boldsymbol{y} = \boldsymbol{y}^o\right) \propto P\left(\boldsymbol{y} = \boldsymbol{y}^o \middle| \boldsymbol{x} = \boldsymbol{x}_t\right) P(\boldsymbol{x} = \boldsymbol{x}_t), \tag{16.9}$$

where the superscript t represents the "true" solution and o represents observed value. Thus (16.9) defines an N-dimensional PDF, which is denoted as $P_a(\boldsymbol{x})$, where a represents the analysis. Lorenc then tells us that the "best" estimate, $\boldsymbol{x}_a$, is either the mean of $P_a(\boldsymbol{x})$,

$$\boldsymbol{x}_a = \int \boldsymbol{x} P_a(\boldsymbol{x})\, d\boldsymbol{x}, \tag{16.10}$$

or the mode, which is

$$\boldsymbol{x}_a = \boldsymbol{x} \text{ such that } P_a(\boldsymbol{x}) \text{ is maximum.} \tag{16.11}$$

As mentioned earlier, the mean and the mode are the minimum variance and the maximum likelihood states, respectively.

16.2.1 Gaussian Framework

We now consider the probability, $P(\boldsymbol{x} = \boldsymbol{x}_t)$, that represents our knowledge about $\boldsymbol{x}$ before the observations are taken. This can be considered as an error, $\boldsymbol{\varepsilon}_b$, which we define as

$$\boldsymbol{\varepsilon}_b \equiv \boldsymbol{x} - \boldsymbol{x}_b, \tag{16.12}$$

where $\boldsymbol{x}_b$ represents the background state, and thus we are considering deviations away from this state, and has a probability $P_b(\boldsymbol{x} - \boldsymbol{x}_b)$, which will be defined soon, once the rest of the problem is set up.

The observational error can be written in terms of two parts [259]. The first part is due to instrumental errors and the second part is the representativeness error. We shall consider these as one entity as the total observational error. This then enables us to write the conditional part of the probability of (16.9) as

$$P\left(\boldsymbol{y} = \boldsymbol{y}^o \middle| \boldsymbol{x} = \boldsymbol{x}_t\right) = P_o\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right), \tag{16.13}$$

which is the observational error, $\boldsymbol{\varepsilon}_o$. We have assumed that the observational and the background errors are independent, which is an acceptable assumption [259], and is made in most data assimilation schemes. Combining (16.12) and (16.13) enables us to write (16.9) as

$$P_a(\boldsymbol{x}) = P_o\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right) P_b(\boldsymbol{x} - \boldsymbol{x}_b) \equiv P_o(\boldsymbol{\varepsilon}_o) P_b(\boldsymbol{\varepsilon}_b). \tag{16.14}$$

We now define these probabilities in terms of a multivariate Gaussian, MG, PDFs such that $\boldsymbol{\varepsilon}_o \sim MG(\mathbf{0}, \mathbf{R})$ in other words the observational errors are multivariate Gaussian distributed with vector of means $\mathbf{0}$ and error covariance matrix, $\mathbf{R}$. The background errors are such that $\boldsymbol{\varepsilon}_b \sim MG(\mathbf{0}, \mathbf{B})$; that is to say that the background errors are also multivariate Gaussian distributed, with the same mean vector, but with error covariance matrix $\mathbf{B}$. The distributions mentioned above are defined by

$$P_b(\boldsymbol{\varepsilon}_b) \propto \exp\left\{-\frac{1}{2}\boldsymbol{\varepsilon}_b^T B^{-1} \boldsymbol{\varepsilon}_b\right\} \equiv \exp\left\{-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_b)^T \mathbf{B}^{-1}(\boldsymbol{x} - \boldsymbol{x}_b)\right\}, \tag{16.15}$$

$$P_o(\boldsymbol{\varepsilon}_o) \propto \exp\left\{-\frac{1}{2}\boldsymbol{\varepsilon}_o^T R^{-1} \boldsymbol{\varepsilon}_o\right\} \equiv \exp\left\{-\frac{1}{2}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right)^T \mathbf{R}^{-1}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right)\right\}. \tag{16.16}$$

Substituting (16.15) and (16.16) into (16.9) yields

$$P_a(\boldsymbol{x}) \propto \exp\left\{-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_b)^T \mathbf{B}^{-1}(\boldsymbol{x} - \boldsymbol{x}_b) - \frac{1}{2}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right)^T \mathbf{R}^{-1}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right)\right\}. \tag{16.17}$$

To maximize P_a is equivalent to minimizing $-\ln$ of (16.17). This results in the nonlinear cost function, J,

$$J(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}_b)^T \mathbf{B}^{-1}(\boldsymbol{x} - \boldsymbol{x}_b) + \frac{1}{2}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right)^T \mathbf{R}^{-1}\left(\boldsymbol{y}^o - \boldsymbol{h}(\boldsymbol{x})\right). \tag{16.18}$$

If we consider an unconstrained minimization method, such as the nonlinear conjugate gradient or quasi-Newton methods, to find the minimum of (16.18), then we require the Jacobian and the Hessian of (16.18). The Jacobian vector of (16.18) can easily be verified as

$$\frac{\partial J}{\partial \boldsymbol{x}} \equiv \mathbf{B}^{-1}\left(\boldsymbol{x}-\boldsymbol{x}_b\right)-\mathbf{H}^T\mathbf{R}^{-1}\left(\boldsymbol{y}-\boldsymbol{h}\left(\boldsymbol{x}\right)\right), \tag{16.19}$$

where $\mathbf{H}$ is defined as

$$\mathbf{H}=\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}, \tag{16.20}$$

and is the Jacobian matrix of $\boldsymbol{h}$ with dimensions $N_o \times N$ and $\dfrac{\partial J}{\partial \boldsymbol{x}}$ has dimensions $N \times 1$ where we have dropped the superscript o for the rest of the chapter as we are now just dealing with the physical observations.

For the Hessian of (16.18) we present this componentwise, but be aware that these entries are components that form a series of matrix multiplications. Thus the components of the Hessian matrix are given by

$$\frac{\partial^2 J}{\partial x_i \partial x_j} \equiv \left[\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right]_{ij}-\left[\mathbf{G}_i\mathbf{R}^{-1}\left(\boldsymbol{y}-\boldsymbol{h}\left(\boldsymbol{x}\right)\right)\right]_j, \tag{16.21}$$

where G is the Hessian of $\boldsymbol{h}$ with

$$\mathbf{G}_i \equiv \frac{\partial}{\partial x_i}\left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right). \tag{16.22}$$

Therefore the dimensions of the full Hessian matrix of J is $N_o \times N$ and where there are N of the G_i matrices with $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, N_o$.

Everything presented above is for what is referred to as the **full field** formulation. However, while Lorenc does not introduce incremental VAR in [259], he does linearize the problem. The linearization is associated with the observation operator $\boldsymbol{h}\left(\boldsymbol{x}\right)$, which is based on being able to approximate the observation operator at the true state by the observation operator at the background state plus a small perturbation, which implies

$$\boldsymbol{h}\left(\boldsymbol{x}_a\right)=\boldsymbol{h}\left(\boldsymbol{x}_b\right)+\mathbf{H}\delta\boldsymbol{x}, \tag{16.23}$$

where $\mathbf{H}$ is the Jacobian of the observation operator, and where $\boldsymbol{x}_a$ is the state that minimizes J, and is the nonlinear solution to

$$\mathbf{0}=\mathbf{H}^T\left(\mathbf{O}+\mathbf{F}\right)^{-1}\left(\boldsymbol{y}-\boldsymbol{h}\left(\boldsymbol{x}_a\right)\right)+\mathbf{B}^{-1}\left(\boldsymbol{x}_b-\boldsymbol{x}_a\right). \tag{16.24}$$

We now assume that the linearization in (16.23) is valid in the entire range of probable values of $\boldsymbol{x}$ in the region of $\boldsymbol{x}_b$, as such an explicit solution can be found by evaluating $\boldsymbol{h}$ at $\boldsymbol{x}=\boldsymbol{x}_b$, and rewriting (16.23) as

$$\boldsymbol{h}\left(\boldsymbol{x}_a\right)=\boldsymbol{h}\left(\boldsymbol{x}_b\right)+\mathbf{H}\left(\boldsymbol{x}_b-\boldsymbol{x}_a\right). \tag{16.25}$$

Through substituting (16.25) into (16.24), it can be shown that the following solutions are equivalent:

$$\boldsymbol{x}_a = \boldsymbol{x}_b + \left(\mathbf{B}\mathbf{H}^T \left(\mathbf{O}+\mathbf{F}\right)^{-1}\mathbf{H} + \mathbf{I}\right)^{-1} \mathbf{B}\mathbf{H}^T \left(\mathbf{O}+\mathbf{F}\right)^{-1} \left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right), \tag{16.26a}$$

$$\boldsymbol{x}_a = \boldsymbol{x}_b + \mathbf{B}\mathbf{H}^T \left(\mathbf{O}+\mathbf{F}\right)^{-1} \left(\mathbf{H}\mathbf{B}\mathbf{H}^T \left(\mathbf{O}+\mathbf{F}\right)^{-1} + \mathbf{I}\right)^{-1} \left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right), \tag{16.26b}$$

$$\boldsymbol{x}_a = \boldsymbol{x}_b + \mathbf{B}\mathbf{H}^T \left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{O} + \mathbf{F}\right)^{-1} \left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right). \tag{16.26c}$$

Exercise 16.3. *Show that the three expressions in* (16.26a)–(16.26c) *are equivalent.*

Finally, in the derivation section of [259], Lorenc refers to the *expected analysis error covariance for this linearized Gaussian case* as given by

$$\left\langle \left(\boldsymbol{x}_a - \boldsymbol{x}_b\right)\left(\boldsymbol{x}_a - \boldsymbol{x}_b\right)^T \right\rangle = \mathbf{B} - \mathbf{B}\mathbf{H}^T \left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{O} + \mathbf{F}\right)^{-1} \mathbf{H}\mathbf{B}. \tag{16.27}$$

However, note that the technique followed here is to find the maximum likelihood state, not the minimum variance state, which is the case for the optimum interpolation approaches. Also recall that for Gaussian distributions three descriptive statistics, mode, median, and mean, are equal.

Two of the three expressions in (16.26a)–(16.26c) are in model space, while one of them is in observation space. An important feature to note about the model space-based solutions is that it is possible to define these solutions in spectral space, or in grid point space, if we are using a spectral model. This was the case at the National Meteorological Center (NMC), which is now the National Centers for Environmental Prediction, NCEP, where their first 3D VAR system was referred to as the **Spectral Statistical Interpolation (SSI)** [328], whereas the later grid point versions was called the **Gridpoint Statistical Interpolation (GSI)** [219].

What we have to consider here is that in 3D VAR, or non-temporal-based variational schemes, we are seeking the initial conditions at set times that minimize the differences between the model and the observations, subject to uncertainties in the model state and the observations. We have provided a schematic of the series of 3D VAR analysis steps in Fig. 16.2. We have mentioned that these techniques are referred to as **filtering**, as we are filtering the scales at these analysis times. The process of continually applying the filtering, and the smoothing in 4D VAR, is referred to as **cycling**. This raises the question of; how long should the time be between analysis updates? We should not that the answer will be geophysically dependent. We have to notice that we are **not** compensating for any model error when using 3D VAR, and the only information we have to compensate for the model error comes from the observations we have at, or near, the analysis time, but these are also not perfect.

In Fig. 16.2 we have drawn an illustration of a situation where if we leave the time between analysis time too long then it is highly likely that the observations would be reject in quality control as they are too far away from the background state, therefore the length of time between cycles should be less than the time scales associated with the model error.

3D VAR is also sensitive to the background and observational error covariance matrices; we shall go into more detail about different models that are used for these matrices in Chapter 17, and can affect the ability of the scheme to converge or to assimilate certain observations. To overcome the lack of temporal information to constrain the model that is a downside to 3D VAR, there was much development to include the time component as Sasaki had indicated in [386] and as such we now move on to consider the development that leads to the 4D VAR cost function.

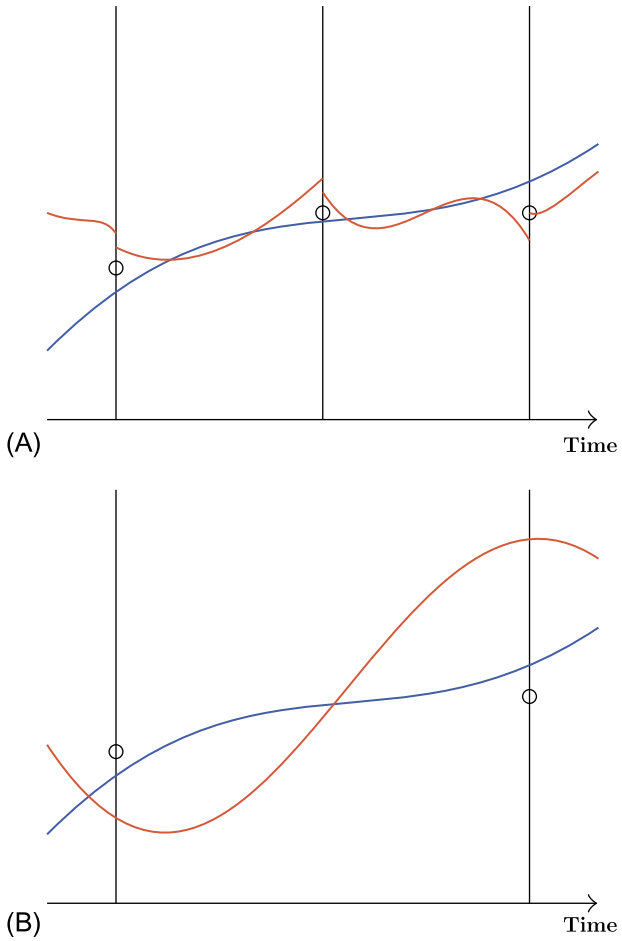


FIGURE 16.2

(A) Schematic of 3D VAR filtering when the cycling length is short enough to control model error, where the blue line is the true trajectory and the red lines are the 3D VAR analysis trajectories. (B) Schematic of 3D VAR filtering when the cycling length is too long to control model error, where the blue line is the true state and the red line is the 3D VAR analysis trajectory, and the circles are observations. The circles are observations.

16.3 Four-Dimensional Data Assimilation

The earliest form of the mathematical ideas behind modern 4D VAR appears in Lewis and Derber [251]. The approach described in [251] is to find the minimum of the cost function

$$J\left(\boldsymbol{x}\left(t_0\right)\right)=\frac{1}{2}\sum_{i=0}^{t_a}\left\langle \mathbf{W}\left(t_i\right)\left(\boldsymbol{x}\left(t_i\right)-\boldsymbol{x}_b\left(t_i\right)\right),\boldsymbol{x}\left(t_i\right)-\boldsymbol{x}_b\left(t_i\right)\right\rangle, \tag{16.28}$$

where t_a is the analysis time, $\mathbf{W}$ is a *weight* matrix that can be changed depending on known accuracies, the expression $\langle \cdot, \cdot \rangle$ is the inner product operator, $\boldsymbol{x}_b$ is the output from a numerical model that has been started by some set of initial conditions, $\boldsymbol{x}_{b,0}$ and $\boldsymbol{x}$ is the analyses that has come from a simpler version of data assimilation, i.e., optimum interpolation (OI). The problem is to seek the initial conditions that minimize the weighted squared differences between the original analysis from the OI scheme at several times and the coincident solutions to the numerical model. Note: In later formulations $\mathbf{W}$ becomes the background error covariance matrix, $\mathbf{B}$.

The minimum of (16.28) is found through its gradient, ∇J, with respect to the initial conditions. To find the minimum of (16.28), an adjoint approach is used. This approach starts by considering the first-order change to (16.28) from a small perturbation, $\boldsymbol{x}'(t_0)$, about the initial conditions $\boldsymbol{x}(t_0)$. Therefore, J' is the first-order change in the functional and is related to the directional derivative in $\boldsymbol{x}'(t_0)$ by

$$J' = \langle \nabla J, \boldsymbol{x}'(t_0) \rangle. \tag{16.29}$$

Substituting the information from (16.28) into (16.29) and introducing a linearized perturbation model, the reader is referred to [251] for more details about this, and through using the property of adjoints,

$$\langle \boldsymbol{x}, \mathbf{A}\boldsymbol{y} \rangle = \left\langle \mathbf{A}^T \boldsymbol{x}, \boldsymbol{y} \right\rangle, \tag{16.30}$$

then the gradient can be expressed as

$$\nabla J = \sum_{i=0}^{t_a} \left(\left[\prod_{k=0}^{K_i} \mathbf{D}^T (t_0 + k\Delta t) \right] \hat{\boldsymbol{x}}_i (t_i) \right), \tag{16.31}$$

where $\mathbf{D}$ is the matrix containing the coefficient of the discrete approximation to the linearized perturbation equation and K_i represents the total number of time steps from t_0 to t_i, where $i = 1, 2, \ldots, K$, and K is the total number of time steps to t_a.

The extension of the adjoint techniques from Lewis and Derber (1985) to an observational-based approach appears in Le Demit and Talagrand [245]. The approach in [245] is based upon calculus of variations techniques and optimal control theory. The starting point in [245] is the state vector, $\boldsymbol{x}$, that is defined in time by the equation

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{M}(\boldsymbol{x}), \tag{16.32}$$

where $\mathcal{M}$ is a continuous nonlinear model operator acting on $\boldsymbol{x}$.

It is assumed that there are sets of observations at different times, denoted by $\boldsymbol{y}(t_1), \boldsymbol{y}(t_2), \ldots, \boldsymbol{y}(t_a)$. Given these observations, it is possible to define a functional in terms of the observations as

$$J(\boldsymbol{x}(t)) = \sum_{i=1}^{t_a} \langle \boldsymbol{x}(t_i) - \boldsymbol{y}(t_i), \boldsymbol{x}(t_i) - \boldsymbol{y}(t_i) \rangle. \tag{16.33}$$

However, there is the problem that the observations do not exactly match the state vector and therefore the gradient of (16.33) cannot be assumed to be exactly zero.

As in [251], it is the initial conditions to (16.32), such that (16.33) is minimized, that are required. To achieve this goal we take the first variation of (16.33) with respect to a small perturbation to the

initial conditions, $\delta \boldsymbol{x}(t_0)$. Following the techniques summarized above with the continuous perturbation equation given by

$$\frac{d\delta \boldsymbol{x}}{dt} = \mathbf{A}(t)\,\delta \boldsymbol{x}, \tag{16.34}$$

where (16.34) is started from initial conditions $\delta \boldsymbol{x}(t_0)$ and

$$\mathbf{A}(t) = \frac{\partial \boldsymbol{\mathcal{M}}(\boldsymbol{x}(t))}{\partial \boldsymbol{x}(t)},$$

is the Jacobian matrix of the model equations, then the gradient is given by

$$\nabla J = 2\sum_{i=0}^{t_a} \mathbf{L}^*(t_i, t_0)\left(\boldsymbol{x}(t_i) - \boldsymbol{y}(t_i)\right), \tag{16.35}$$

where the property that (16.34) is a linear equation for $\delta \boldsymbol{x}$ has been used, and therefore the solution at $t = t_i$ depends linearly on $\delta \boldsymbol{x}(t_0)$. The linear operator in (16.35) is referred to as the **resolvent** between t_0 and t_i, denoted by $\mathbf{L}(t, t_0)$, and $\mathbf{L}^*(t_i, t_0)$ is its **adjoint**.

The more advance observation operator version of (16.33) appears in Talagrand [422], where this cost function is

$$J(\boldsymbol{x}(t)) = \frac{1}{2}\sum_{i=1}^{t_a} \langle \boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}(t_i)), \boldsymbol{y}(t_i) - \boldsymbol{h}_i(\boldsymbol{x}(t_i))\rangle. \tag{16.36}$$

Given that the observation operator is a function of the model state at t_i, then a discrete version of the dynamical model equations is required, which is

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \Delta t\,\boldsymbol{\mathcal{M}}_{0,i}(\boldsymbol{x}_i). \tag{16.37}$$

To find the minimum of (16.36), we require the first variations of (16.36) and (16.37) with respect to the initial conditions in (16.37). Using the techniques summarized above, as well as the adjoint property (16.30), we obtain the expression for the Jacobian of (16.36), given (16.37) as

$$\nabla_{\boldsymbol{x}_0} J = \sum_{i=0}^{t_a} \left(\mathbf{I} + \Delta t \mathbf{M}_0^*\right)\left(\mathbf{I} + \Delta t \mathbf{M}_{0,1}^*\right)\ldots\left(\mathbf{I} + \Delta t \mathbf{M}_{0,i-1}^*\right)\mathbf{H}_i^*\left(\boldsymbol{h}_i(\boldsymbol{x}_i) - \boldsymbol{y}_i\right), \tag{16.38}$$

where

$$\mathbf{H}_i = \frac{\partial \boldsymbol{h}_i(\boldsymbol{x}_i)}{\partial \boldsymbol{x}_i}, \quad \mathbf{M}_i = \frac{\partial \boldsymbol{\mathcal{M}}_{0,i}(\boldsymbol{x}_0)}{\partial \boldsymbol{x}_i},$$

are the tangent linear models of the observation operator and the nonlinear model, respectively, and * is the adjoint operator.

In the three approaches summarized above, the final expressions are independent of the observational error covariance matrix $\mathbf{R}$, nor do they contain a background error term. The observational error

covariance matrix is introduced in Courtier and Talagrand [76]. The background component is speculated upon in Thépaut and Courtier [435] and then applied in Thépaut et al. [436], as an extra constraint in the functional formulation, which leads to

$$J(\boldsymbol{x}_0) = \frac{1}{2}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right) + \frac{1}{2}\sum_{i=1}^{t_a}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right)\right)\right)^T \mathbf{R}_i^{-1}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right)\right)\right). \quad (16.39)$$

The reason to include the background, a priori information is because the problem is ill-posed if we are just considering the fit to the observations as an inverse problem [252,431], as there are not enough observations to constrain all of the degrees of freedom, as such there is no unique solution.

The nonlinear solution to (16.39) is identified through finding the zero of the Jacobian of (16.39), which can be shown to be

$$\nabla_{\boldsymbol{x}_0} J = \mathbf{B}^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_b\right) - \sum_{i=1}^{t_a} \mathbf{H}_i^T \mathbf{M}_{0,t_i}^T \mathbf{R}_i^{-1} \mathbf{d}_i = \mathbf{0}, \quad (16.40)$$

where $\mathbf{d}_i \equiv \left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right)\right)\right)$ is referred to as either the *innovation*, or the **departure vector**.

Exercise 16.4. *Derive the expression in* (16.40) *is equivalent to* (16.39).

Another way of writing (16.40), and the more practical method for coding a **full field 4D VAR** system, is as

$$\nabla_{\boldsymbol{x}_0} J = \mathbf{B}^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_b\right) - \mathbf{M}_1^T \times \left(\mathbf{H}_1\mathbf{d}_1 + \mathbf{M}_2^T\left(\mathbf{H}_2\mathbf{d}_2 + \mathbf{M}_3\left(\cdots + \mathbf{M}_{N_o-1}^T\left(\mathbf{H}_{N_0-1}^T + \mathbf{M}_{N_o}^T\mathbf{H}_{N_o}^T\mathbf{d}_{N_o}\right)\right)\cdots\right)\right), \quad (16.41)$$

where in (16.41) we are moving the innovation at time k back to time $k-1$, and then adding on the scaled innovations $\mathbf{H}_{k-1}^T\mathbf{d}_{k-1}$ to $\mathbf{M}_{k,k-1}^T\mathbf{H}_k^T\mathbf{d}_k$ and so on back to the initial time of the window. Therefore, through programming the gradient in this approach, we see that we only have to evaluate the adjoint of the tangent linear model once through the whole window rather than for each innovation and then collecting their sums at time t_0.

Therefore, in a four-dimensional variational system we are seeking the initial conditions, but we could also be seeking adjusted model parameters as well as bias corrections to the observations, at the start of the assimilation window, given a set of observations through that window. (See Fig. 16.3 for an illustration of the assimilation windows.)

The version of 4D VAR that we have described here is the strong constraint formulation, i.e., no model error. The implication of the strong constraint formulation, as indicated by Sasaki, is that we are not making any corrections to the model, or the trajectory, through the window. Therefore, 4D VAR is the mechanism of finding the initial conditions at the beginning of the window such that the trajectory minimizes the differences between the trajectory and the observations; thus at the end of the window the forecast that is produced is more accurate than the one produced from the previous analysis. Again we have highlighted this in Fig. 16.3.

This raises the question of; how long should the assimilation window be? The answer again is geophysically but also numerically dependent. We must recall that in this formulation we assume that the numerical model is **perfect**. Therefore, the assimilation window should be long enough so that it is economical to be running the assimilation scheme; we should note that we are minimizing a nonlinear least squares problem here, which requires multiple evaluations of the adjoint and the forward nonlinear

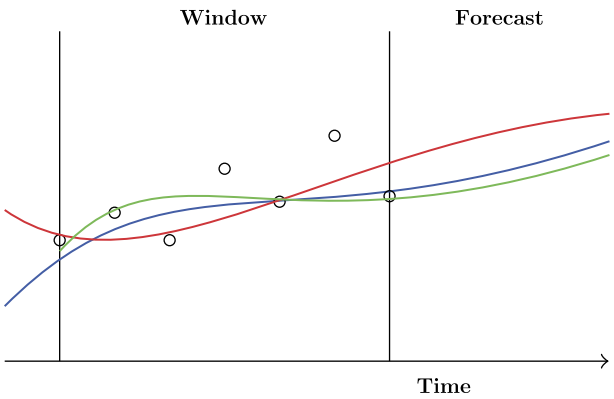


FIGURE 16.3

Schematic of 4D VAR smoothing, where the blue line is the true solution, the red line is the forecast from the previous analysis time and is the background state, and the green line is the analysis trajectory filtering when the cycling length is short enough to control model error, and the circles are observations.

numerical model, which can be computationally expensive. However, we require the window to be short enough to prevent the model error from overwhelming the minimization of the gradient.

In operational synoptic scale numerical weather prediction centers, the window length is typically six hours; this is when there is no compensation for model error. While in an ocean application in the tropical Pacific Ocean, it is shown in [457,468] that it is possible to use a 30-day window for 4D VAR, while in their 3D VAR experiments the authors use 10 days as the distance between analysis cycles. In mesoscale atmospheric applications it may be necessary to only have a window length of a few hours, less than six. However, we can see that to implement full fields 4D VAR is quite expensive if there is no form of preconditioning applied to speed up the decent algorithm or nonlinear iterative solver.

Therefore, there was some reluctance to go to 4D VAR due to these costs until a way around all of the evaluations of the nonlinear model was devised, where this mechanism is referred to as **incremental variational data assimilation**.

16.4 Incremental VAR

As the name of this approach suggests, we are seeking an increment to some geophysical field. Incremental VAR, as it is known now, was introduced by Courtier, Thèpaut, and Hollingsworth at ECMWF in [77], and was a major breakthrough that enabled 4D VAR to be considered operationally viable for the numerical weather and ocean prediction communities [154,357,361,369].

The starting point for the incremental-based variational data assimilation schemes is the assumption that our background state, $\boldsymbol{x}_b$, is a quite good estimate of the true state, $\boldsymbol{x}^t$, and that the two states only differ by a small *increment*, denoted by $\delta\boldsymbol{x}$, such that

$$\boldsymbol{x}^t = \boldsymbol{x}_b + \delta\boldsymbol{x}, \tag{16.42}$$

which can be rearranged to $\delta \boldsymbol{x} = \boldsymbol{x}^t - \boldsymbol{x}_b$. As an aside we should note that this is the linearization that Lorenc introduced to obtain the linear solution in [259], but this was to find the full field linear approximation to the nonlinear problem. We should also note that this is the approach that Tarantola and Valette introduced in [431]. The incremental approach applies to both the spatial 1D, 2D, and 3D VAR schemes as well as to the temporal 4D VAR schemes. We shall show the incremental approach for both the 3D and 4D VAR formulations here.

16.4.1 Incremental Spatial VAR, 1D, 2D, and 3D VAR

In full field 3D VAR we have to minimize the cost function

$$J\left(\boldsymbol{x}^t\right) = \frac{1}{2}\left(\boldsymbol{x}^t - \boldsymbol{x}_b\right)^T \mathbf{B}^{-1}\left(\boldsymbol{x}^t - \boldsymbol{x}_b\right) + \frac{1}{2}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}^t\right)\right)\mathbf{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}^t\right)\right). \tag{16.43}$$

If we look at (16.43) then we see that the only nonlinear term is the observation operator. Therefore, we are going to apply the tangent linear approximation to the observation operator as

$$\boldsymbol{h}\left(\boldsymbol{x}^t\right) \approx \boldsymbol{h}\left(\boldsymbol{x}_b\right) + \mathbf{H}\delta\boldsymbol{x},$$

where $\mathbf{H}$ is again the tangent linear approximation to the observation operator as we have used in the optimum interpolation and the full field 3D VAR schemes. Now if we notice that the background term in (16.42) is our definition for the increment, then we are able to redefine (16.43) as a cost function for the increment $\delta\boldsymbol{x}$ as

$$J\left(\delta\boldsymbol{x}\right) = \frac{1}{2}\left(\delta\boldsymbol{x}\right)^T \mathbf{B}^{-1}\left(\delta\boldsymbol{x}\right) + \frac{1}{2}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right) + \mathbf{H}\delta\boldsymbol{x}\right)^T \mathbf{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right) + \mathbf{H}\delta\boldsymbol{x}\right). \tag{16.44}$$

Now the problem becomes to find the increment, $\delta\boldsymbol{x}^a$, such that (16.44) is minimized. This requires finding the zeros, hopefully only one, of (16.44). Therefore, differentiating (16.44) with respect to $\delta\boldsymbol{x}$ yields

$$\begin{aligned} &\mathbf{B}^{-1}\delta\boldsymbol{x} - \mathbf{H}^T\mathbf{R}^{-1}\left(\mathbf{d}_b - \mathbf{H}\delta\boldsymbol{x}\right) = \mathbf{0}, \\ &\Rightarrow \left(\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)\delta\boldsymbol{x} = \mathbf{H}^T\mathbf{R}^{-1}\mathbf{d}_b, \end{aligned} \tag{16.45}$$

where $\mathbf{d}_b$ is the background innovation, $\boldsymbol{y} - \boldsymbol{h}\left(\boldsymbol{x}_b\right)$. Therefore we have to iteratively find the value of $\delta\boldsymbol{x}$ such that it is possible to invert the matrix equation in (16.45). We should note that we are not updating the background innovation; this is the slight difference to the equation we identified from Tarantola and Valette [430] that we referred to as the nonlinear updated version of incremental 3D VAR. Therefore, to make (16.45) equivalent to a rearrangement of (15.84) we would have to update $\mathbf{d}_b$ with the increment added to $\boldsymbol{x}_b$ and evaluate $\boldsymbol{h}$ and as such $\mathbf{d}_b$. This is not done on each iteration, but in some implications of incremental VAR it is done at a certain stage or stages and this process is referred to as the **inner and outer loops**. We shall go into some details about the inner and outer loops in the next section; but first we move on to incremental 4D VAR.

16.4.2 Incremental Temporal 4D VAR

We have already introduced the concept that made 4D VAR operationally viable, which was to assume that our background state is a good approximation to the true state and that the two states only differ

by a small increment $\delta \boldsymbol{x}$. While this is also true for 4D VAR, we have to recall that we are seeking the optimal initial conditions at the **start** of the assimilation window. Therefore, our increment is now defined as a small change to the background state initial condition as

$$\boldsymbol{x}_0^t = \boldsymbol{x}_{b,0} + \delta \boldsymbol{x}_0 \Rightarrow \delta \boldsymbol{x}_0 = \boldsymbol{x}_0^t - \boldsymbol{x}_{b,0}. \tag{16.46}$$

Thus, we need to find a way to linearize the full field 4D VAR cost function

$$J\left(\boldsymbol{x}_0^t\right) = \frac{1}{2}\left(\boldsymbol{x}_0^t - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}_0^{-1}\left(\boldsymbol{x}_0^t - \boldsymbol{x}_{b,0}\right) + \frac{1}{2}\sum_{i=1}^{N_o}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0^t\right)\right)\right)^T \mathbf{R}_i^{-1}\left(\boldsymbol{y}_i - \boldsymbol{h}\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0^t\right)\right)\right), \tag{16.47}$$

with respect to (16.46). The solution is to apply the tangent linear approximation, used for the observation operator in the incremental 3D VAR derivation, but applied to the nonlinear numerical model as

$$\mathcal{M}_{0,i}\left(\boldsymbol{x}_0^t\right) \approx \mathcal{M}_{0,i}\left(\boldsymbol{x}_{b,0}\right) + \mathbf{M}_{0,i}\delta \boldsymbol{x}_0. \tag{16.48}$$

However, before we are able to implement (16.48) we notice that we have to apply a tangent linear approximation to the composite of two functions. Therefore, in general we have

$$f\left(g\left(x^t\right)\right) = f\left(g\left(x_b + \delta x\right)\right) \approx f\left(g\left(x_b\right)\right) + f'\left(g\left(x_b\right)\right) g'\left(x_b\right) \delta x. \tag{16.49}$$

Given (16.49), then for our application of 4D VAR, we have

$$\begin{aligned} f &= \boldsymbol{h}_i, \\ g &= \mathcal{M}_{0,i}\left(\boldsymbol{x}_0^t\right), \\ f' &= \mathbf{H}_i, \\ g' &= \mathbf{M}_{0,i}. \end{aligned}$$

Thus, if we defined the ith innovation vector as $\mathbf{d}_{b,i} \equiv \boldsymbol{y}_i - \left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_{b,0}\right)\right)$, then we can linearize (16.47) as

$$J\left(\delta \boldsymbol{x}_0\right) = \frac{1}{2}\left(\delta \boldsymbol{x}_0\right)^T \mathbf{B}_0^{-1}\left(\delta \boldsymbol{x}_0\right) + \frac{1}{2}\sum_{i=1}^{N_o}\left(\mathbf{d}_{b,i} - \mathbf{H}_i\mathbf{M}_{0,i}\delta \boldsymbol{x}_0\right)^T \mathbf{R}_i^{-1}\left(\mathbf{d}_{b,i} - \mathbf{H}_i\mathbf{M}_{0,i}\delta \boldsymbol{x}_{b,0}\right). \tag{16.50}$$

Therefore, we are now seeking the increment to the initial conditions that minimized (16.50). Thus we require the zeros of the Jacobian of (16.50) with respect to the perturbation to the initial conditions. The Jacobian of (16.50) can easily be shown to be

$$\nabla_{\delta \boldsymbol{x}_0} J = \mathbf{B}_0^{-1}\delta \boldsymbol{x}_0 - \sum_{i=1}^{N_o}\mathbf{M}_{0,i}^T\mathbf{H}_i^T\mathbf{R}_i^{-1}\left(\mathbf{d}_{b,i} - \mathbf{H}_i\mathbf{M}_{0,i}\delta \boldsymbol{x}_0\right) = \mathbf{0}. \tag{16.51}$$

Exercise 16.5. *Verify the Jacobian in* (16.51) *is correct.*

To be able to solve for the zeros of (16.51) we have to apply an iterative solver for the inverse of the matrix-vector equation; however, the difference between the full field and incremental versions of 4D VAR is that now we are only having to evolve the increment throughout the assimilation window that we assume follows the tangent linear approximation introduced in Chapter 13. This hopefully holds as long as the increment is sufficiently small, so that the resolvent does describe its evolution throughout the assimilation window. This is an important feature to note as this puts a constraint on the window length. Thus we require the window length to be of a sufficient size such that the tangent linear approximation holds.

However, incremental 4D VAR has been incredible successful in an operational context, having been implemented in most of the world's operational numerical weather prediction centers in some form. However, as with all variational and non-ensemble-based methods, the need to have flow dependency in the error covariance models is becoming more important as the resolutions of the numerical models increase and as such the assumption for static covariances is not as valid for the smaller dynamical scales; but is some what true for larger synoptic scale errors as well.

The advantage of incremental VAR is its ability to run in lower resolutions that makes the evaluation of the tangent linear model, and the adjoint, much cheaper. However, this has led to much discussion about whether or not the nonlinear trajectory should be updated, which would then update the innovations, and then seek a second estimate of the increment at a higher resolution, which would be linked to the dynamical scale of the geophysical system under consideration. These techniques are referred to as the inner and outer loops. We now explain what these *loops* are.

16.4.3 Inner and Outer Loops

As mentioned over the last two subsections, when we are implementing, or running, incremental VAR we can take advantage of the coarseness of the spatial resolution of the small increment to reduce the cost of running the tangent linear and adjoint models. We now introduce the loose definition of the inner and outer loops in incremental VAR.

Definition 16.6 (Inner Loop). The inner loop refers to a lower spatial resolution, and possible temporal resolution with respect to incremental 4D VAR, that an iterative minimization scheme for finding the minimum of the cost function is run.

Definition 16.7 (Outer Loop). The outer loop refers to a higher-order spatial and temporal resolution, where the nonlinear trajectory of the model in the case of incremental 4D VAR, and the observation operators, and hence the innovations, are updated. This is the case for both incremental 3D and 4D VAR. Note: The iterative scheme is not usually evaluated at this higher resolution. Normally only the nonlinear trajectories and innovations are updated at this resolution.

In incremental 3D VAR we do not have the numerical model to deal with, but we do have nonlinearities in the observation operator. Therefore, in the operational configuration of incremental 3D VAR the advantage of this scheme is that it is possible to run multiple iterations at a lower spatial resolution and then interpolate to a higher resolution, **not necessarily** the full operational model; at this higher resolution it is now possible to run a new observational quality check to see if some of the observations that were rejected in the initial sweep may be acceptable now as the background trajectory may be closer to the observation, and as such this information is then accepted. Now the resolution can either be reduce back to that of the first **inner loop**, a slight reduce resolution to the current loop, or it could be at the updated resolution. We have provided a schematic of a few different configurations of inner loop and outer loops that are possible with incremental 3D VAR in Fig. 16.4.

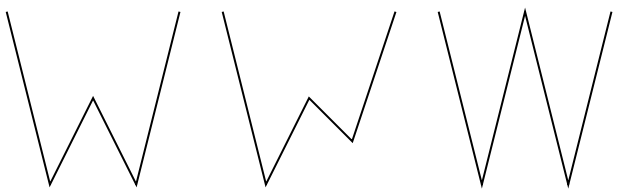


FIGURE 16.4

Schematics of different possible configurations for the resolution of the inner and outer loops in 3D VAR.

With respect to incremental 4D VAR, we have to take into account that we are running the tangent linear and the adjoint models at the resolution of the inner loops and as such it becomes more expensive to run these data assimilation schemes at higher resolutions. There was much debate after incremental 4D VAR become operationally viable about how many, if any, outer loops needed to be evaluated to ensure that we have correctly resolved the scales that are important to ensure that it is possible to produce the best forecast. The ECMWF were the first to run an incremental 4D VAR scheme with **two** outer loops. This means that they were running the full nonlinear model twice during the minimization process. Now this does not mean that they were running the nonlinear model at the full forecast resolution, which indicates a trick that incremental VAR can do. In this updated second edition, ECMWF runs three outer loops.

The point of the outer loop is to provide **some** nonlinear information back into the minimization scheme, especially if the increment has move towards the limit of the viability of the tangent linear approximation about the current nonlinear trajectory. Of course the other advantage of performing an outer loop is the possibility of introducing new observations into the minimization, which could aid in the analysis.

16.4.4 Nonlinearities and Outer Loops

Since the first edition of this textbook, there are has been a detailed study published in 2018 that looked at the effects of nonlinearities in 4D VAR, [45]. As stated earlier, in the incremental form of variational data assimilation we introduce the tangent linear approximation to the observation operator in 3D VAR, but to the composition of the observation operator with the nonlinear model in 4DVAR. However, if we consider a Taylor series expansion of the component of 4D VAR then to third order we have

$$\begin{aligned}
&\mathbf{y}_k - \mathbf{h}_k\left(\mathcal{M}_{0,t_k}\left(\mathbf{x}_0\right)\right)\mathbf{y}_k - \mathbf{h}_k\left(\mathcal{M}_{0,t_k}\left(\mathbf{x}_0^g + \delta\mathbf{x}_0\right)\right) \\
&\qquad = \mathbf{y}_k - \mathbf{h}_k\left(\mathcal{M}_{0,t_k}\left(\mathbf{x}_0^g\right)\right) - \mathbf{H}_k\mathbf{M}_{o,t_k}\left(\delta\mathbf{x}_0\right) - \frac{1}{2}\left(\delta\mathbf{x}_0\right)^T\left(\frac{\partial\mathbf{H}_k\mathbf{M}_{0,t_k}}{\partial\mathbf{x}}\right)\left(\delta\mathbf{x}_0\right) - O\left(||\delta\mathbf{x}_0||^3\right), \\
&\qquad \approx \mathbf{y}_k - \mathbf{h}_k\left(\mathcal{M}_{0,t_k}\left(\mathbf{x}_0^g\right)\right) - \mathbf{H}_k\mathbf{M}_{o,t_k}\left(\delta\mathbf{x}_0\right), \qquad (16.52)
\end{aligned}$$

where $\mathbf{x}_0^g$ is referred to as the guess trajectory that the cost function is linearized around. When considering the Hessian of the linearized cost function there is an extra term that appears in the form a set of vector differentiation that creates a series of matrices. This term is ignored here, thus reducing to the tangent linear approximation.

The validity of the tangent linear approximation is based upon whether either the increment $\delta \mathbf{x}_0$ are in some sense small or the dependence of the linearization of $\mathbf{h}_k\left(\mathcal{M}_{0,t_k}\left(\mathbf{x}_0\right)\right)$ on the reference trajectory is negligible. It is stated in [45] that concerning the first aspect, they note that the size of analysis increments is, to first order, a linear function of observation departures. Thus, the sizes of departures need to be small with respect to the observation and background errors used in the analysis update for the tangent linear approximation to hold. The other aspect affecting the validity of the tangent linear approximation relies on an **implicit linearity assumption of both the forecast model and the observation operator in a neighborhood of the reference trajectory**.

It is then highlighted that experience at ECMWF indicates a clear sensitivity of both the linearized observation operator and the linearized model to the linearization state. Given these concerns [45] revisits the roles of model and observation nonlinearities in the current operational ECMWF 4D-Var implementation and to validate the effectiveness of the incremental 4D-Var method in dealing with these nonlinearities.

With respect to the nonlinear numerical prediction model, [45] indicate that the nonlinearities here affect the 4D VAR solution in two main ways. First, the more nonlinear the high-resolution trajectory solution is, the spatially noisier the low-resolution interpolated linearization state for the 4D VAR inner loops becomes. This roughness of the interpolated trajectory increases when differences between the time steps and resolutions of the inner loops and the trajectory become larger. Second, the tangent linear evolution differs more from the nonlinear solution as nonlinearities increase. One measure of the degree of nonlinearity, which comes from [356], is to take the difference between the nonlinearly and linearly evolved increments in the last minimization,

$$\mathcal{M}\left(\mathbf{x}^{n\text{-}1}+\delta \mathbf{x}^n\right)-\left(\mathcal{M}\left(\mathbf{x}^{n-1}\right)+\mathbf{M}\delta \mathbf{x}^n\right). \tag{16.53}$$

Over the years at ECMWF, there has been an increase in the resolution of the trajectory and the inner loops and the gap in resolution between the two has increased. This has resulted in increased differences, which is interpreted as increased nonlinearity due to the combination of increased model resolution and resolution differences between the inner loop and the trajectory. It is stated that one way to counteract the nonlinearity coming from the resolution increases is to shorten the length of the assimilation window. This can be achieved either by very short windows or by the use of overlapping assimilation windows. In this second case the reduction in nonlinearity is realized by reducing the size of the analysis increments, $\delta \mathbf{x}$, as each new window will start from a first guess trajectory that has already seen the observations in the overlapped part of the window.

Another form of nonlinearity comes from the observation operators, and measure suggested in [45] comes from the ECMWF's Ensemble of Data Assimilations (EDA). We do not explain what this is here, as it is part of the ensemble chapter. The reader is referred to [45] for the details about these statistics.

The part of [45] that we wish to highlight is the approach for dealing with nonlinearities. The incremental approach to 4D VAR reduces the resolution of the inner loops to make the solution more affordable. Observation departures are calculated at high resolution and then the high-resolution trajectory is truncated and interpolated to the resolution of the inner loop for each time step of the low-resolution minimization, as shown earlier. At the end of the minimization, the increments are projected back to the high resolution and added to the previous trajectory at the start of the assimilation window. This process is repeated for all minimizations, which can be at different resolutions, **starting with the lowest resolution to capture the larger scales and increasing the resolution in later minimizations to extract more detailed information from the observations**.

In [45] there are a few diagnostics to determine the impacts of the nonlinearities, and the one we present here is for model space. An indicator of the success or otherwise of an incremental strategy is the size of the analysis increments produced by the linearized cost function during successive outer loop iterations. It is stated in [45] that for a well-behaved incremental 4D VAR converging towards the solution of the nonlinear cost function, successive analysis increments are expected to become smaller, reflecting the hypothesis that successive first guess trajectories provide increasingly accurate descriptions of the flow. [45] presents experimental results, and we have a copy of Figure 6 from [45] in Fig. 16.5, where they present the vertical profiles of the standard deviations of the analysis increments of vorticity (left panel) and temperature (right panel) from a multi-incremental 4D-Var experiment with five outer loops (in this experiment the outer loop resolution is TCo399, approx. 30 km, and the inner loop resolutions are TL95/TL159/TL255/TL255/TL255, approx. 210, 125, and 80 km).

In this figure the plots are vertical profiles of the globally averaged standard deviation of the analysis increments produced by successive outer loop iterations for vorticity (a) and temperature (b). Values have been averaged over a 1-month period. The assimilation experiment has been run with an outer loop resolution corresponding to a cubic octahedral reduced Gaussian grid with spectral truncation 399 (TCo399, approx. 30 km grid spacing), and inner loop resolutions corresponding to linear reduced resolution Gaussian grids at spectral truncations TL95/159/255/255/255, corresponding to approx. 210/120/80 km grid spacing.

In [45] it is stated that the magnitude of the analysis increments is seen to gradually decrease for successive outer loop iterations, more rapidly in the stratosphere for vorticity. After **five outer loop** iterations, the magnitude of the analysis increments appears to asymptote to a relatively small value for temperature throughout the atmospheric column ($\Delta T_a \approx 0.05$ K), and for vorticity in the stratosphere and mesosphere ($\Delta vo_a \approx 10 - 7s - 1$ for model levels greater than 70). However, it appears that incremental 4D VAR does not seem to have fully converged for vorticity in the troposphere. This is confirmed by the longitudinal averages of the analysis increments produced by the first and last outer

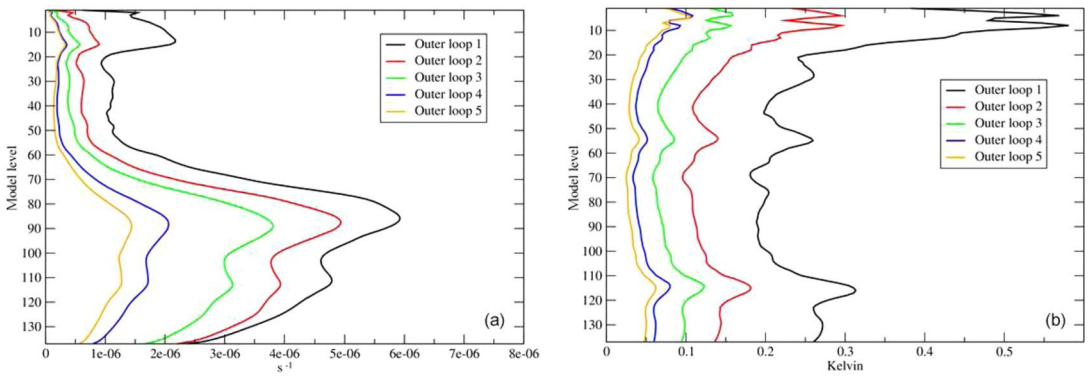


FIGURE 16.5

Copy of figure 6 from Bonavita, M., Lean, P., and Holm, E.: Nonlinear effects in 4D-Var, Nonlin. Processes Geophys., 25, 713–729, https://doi.org/10.5194/npg-25-713-2018, 2018. https://creativecommons.org/licenses/by/4.0/.

loops for temperature, vorticity and humidity, that are shown in Fig. 7 from [45]. We have a copy of this in Fig. 16.6.

It is apparent how the last outer loop iteration still manages to produce non-negligible increments for the tropospheric wind and humidity fields (middle and bottom rows in Fig. 16.6), as a result of the increased presence of nonlinear observations and the increased nonlinearity of the relevant meteorology, where it is suggested this could be due to organized convection and baroclinic instability. [45] states that this is suggesting that increasing the number of outer loops from the current operational value of three up to at least five can lead to a better use of available observations and, ultimately, more accurate analyses and forecasts.

As an aside [45] indicates and interesting aspect of this investigation has been to highlight the relatively large analysis increments produced by 4D VAR in the mesosphere, above model level 20 in Fig. 16.6 and suggests that this is due to a combination of relatively inaccurate model dynamics due to sponge layer effects and the scarcity of observational constraints in this part of the atmosphere, where [45] highlights that the highest peaking channels from current microwave sounders are only marginally sensitive to this upper atmospheric layer.

The reader is referred to [45] for some more results, as well as very interesting discussions. This is a well laid out paper, and it very clear to follow.

We now move on to consider another technique to deal with optimizing the use of observations to constrain the data assimilation systems.

16.4.5 First Guess at Appropriate Time

As mentioned at the beginning of this chapter, there is a variational scheme that has been used at operational numerical prediction centers that is referred to as FGAT. This technique is similar to the incremental 4D VAR formulation but instead of having the model move the increment to the time of the observations we have the identity matrix [242] and as such the cost function for FGAT is given by

$$J(\delta\mathbf{x}_0) = \frac{1}{2}\delta\mathbf{x}_0^T\mathbf{B}_0^{-1}\delta\mathbf{x}_0 + \frac{1}{2}\sum_{i=-\frac{N}{2}}^{\frac{N}{2}}(\mathbf{d}_i - \mathbf{H}_i\delta\mathbf{x}_0)^T\mathbf{R}_i^{-1}(\mathbf{d}_i - \mathbf{H}_i\delta\mathbf{x}_0), \tag{16.54}$$

where the analysis conditions are calculated at the mid-point in time between the two cycling times. In [242] Lawless examines the accuracy of this formulation of FGAT for the analysis errors and finds in the toy case that he considers the analysis error variance could be larger than the variance of the inputs.

An alternative approach to FGAT is to not interpolate the initial conditions to the observation times, but rather to use an increment from the model at the observation times, and interpolate them to the analysis time.

In [268] Lorenc and Rawlins provide a very good schematic to illustrate the differences between 3D VAR, 3D-FGAT and 4D VAR, and we have a copy of that diagram in Fig. 16.7.

As great as incremental VAR is with respect to making the assimilation of observations in time, it is still computationally expensive to run compared to 3D VAR; we should note that nearly every implementation of incremental 4D VAR at operational centers has beaten their previous either full field or incremental 3D VAR system and some thoughts on why this is so are presented in [268]. However, the wish of centers that have implemented incremental 4D VAR is the ability to **increase** the window length. This implies that we need to address the problem of model error.

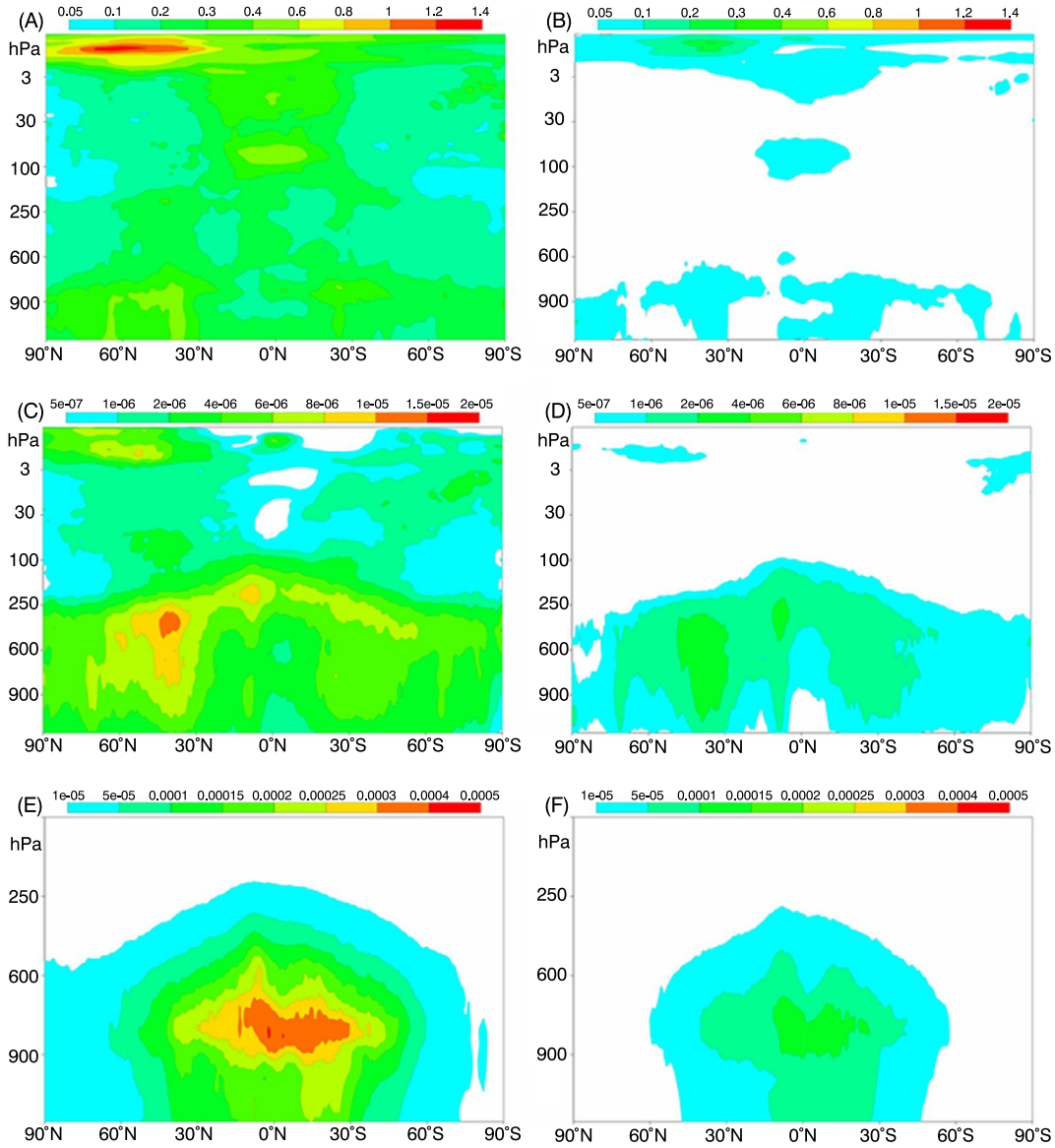


FIGURE 16.6

Copy of figure 7 from Bonavita, M., Lean, P., and Holm, E.: Nonlinear effects in 4D-Var, Nonlin. Processes Geophys., 25, 713–729, https://doi.org/10.5194/npg-25-713-2018, 2018. https://creativecommons.org/licenses/by/4.0/.

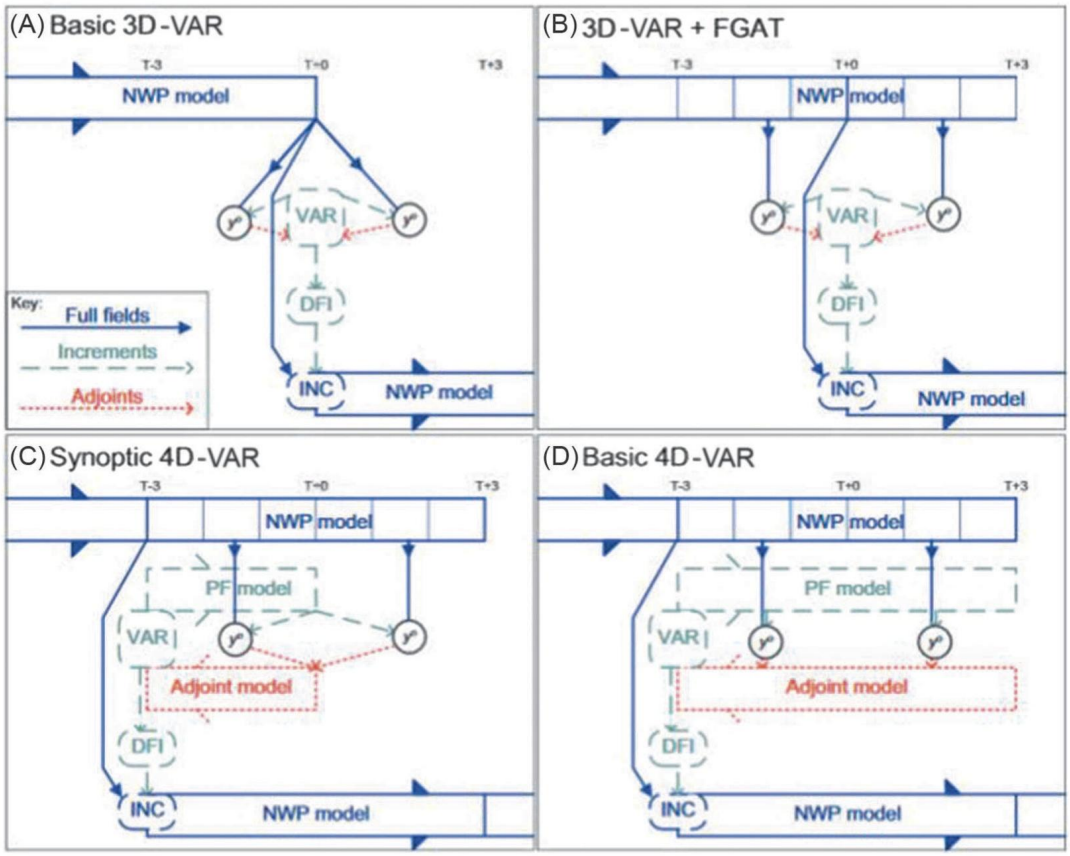


FIGURE 16.7

Copy of Figure 1 showing the different versions of VAR considered from [268]. Note we have not explained the synoptic 4D VAR, as that was part of the testing in the paper.

16.5 Weak Constraint—Model Error 4D VAR

We recall that the term **weak-constraint** was introduced by Sasaki in 1970 [386] in Section 16.1, where the main idea of this formulation is the assumption that the model equations are not exact; therefore it is sufficient to only approximately satisfy them. As stated in [442], one of the problems of implementing the weak constraint formulation is the computational cost, but also the lack of information with which to define the mode error covariance matrix which is required to solve the problem.

In Yannick Trèmolet's 2006 paper [442], he considers three different formulations of model error control variables for 4D VAR, where control variables are transforms of the model state that we wish to analyze such that the resulting transform problem is either of a lower number of analysis variables, and/or less correlated variables. Trèmolet defines three different control variables: model-bias control variable, model-error forcing control variable, and model-state forcing control variable.

Following [442], then

> Model error can be defined as the difference between the perfect model trajectory and the state at each time step over the length of the assimilation window for a given initial condition.

Mathematically, if we define the model error parameter by $\boldsymbol{\beta} = \{\boldsymbol{\beta}_i\}$ for $i = 1, 2, \ldots, n$, where n represents the total number of time steps in the assimilation window, then β satisfies

$$\boldsymbol{x}_i = \mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right) + \boldsymbol{\beta}_i, \tag{16.55}$$

where $\mathcal{M}_{0,i}$ represents the nonlinear model from time $t = t_0$ to $t = t_i$ and $\boldsymbol{x}_0$ are the initial conditions.

16.5.1 Model-Bias Control Variable

Given the expression in (16.55) for the model error, we can define the constraint associated with this formulation as

$$F_i\left(\boldsymbol{x}_i\right) = \boldsymbol{x}_i - \mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right). \tag{16.56}$$

However, as stated in [442], the configuration of the ECMWF global synoptic numerical weather prediction model has 10^9 variables, and as such the error covariance matrix for the model error would be $10^9 \times 10^9$ which would contain 10^{18} elements, but we have to realize that we need to evaluate this matrix at **every** time step. Therefore, to simplify things we shall assume that the bias is **constant** throughout the assimilation window.

Therefore the cost function associated with this formulation of the model error is given by

$$\begin{aligned} J\left(\boldsymbol{x}_0, \boldsymbol{\beta}\right) = & \frac{1}{2}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}_0^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right) + \frac{1}{2}\boldsymbol{\beta}^T \mathbf{Q}^{-1}\boldsymbol{\beta} \\ & + \frac{1}{2}\sum_{i=1}^{N_o}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right) + \boldsymbol{\beta}\right)\right)^T \mathbf{R}_i^{-1}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0\right) + \boldsymbol{\beta}\right)\right), \end{aligned} \tag{16.57}$$

where $\mathbf{Q}$ is the **model error covariance matrix**. As always we are assuming that the background, observational, and model, errors are independent of each other as well as that the observational errors are uncorrelated in time.

16.5.2 Modeling the Model Error Covariance Matrix

One of the other drawbacks of including the model error component into the cost function is the need of an estimate for the model error covariance matrices, $\mathbf{Q}_i$. For the bias model error case we require the one estimate for $\mathbf{Q}$, whereas for the other two approaches we required an estimate for this matrix for each interval.

Trèmolet [443] discusses different approaches that were tried at ECMWF for modeling the $\mathbf{Q}$ matrix. The first approach is to assume that the model error covariance matrix is a factor of the background error covariance matrix; that is to say, $\mathbf{Q} = \alpha\mathbf{B}_0$. The problem with this approximation is that it constrains the initial condition increment, $\delta\boldsymbol{x}_0$, and the model error increment, $\delta\boldsymbol{\eta}$, to the same direction. Therefore, the model error is restricted to the same subspace as the initial conditions increments, where only their relative amplitudes are different.

This approach appears to introduce two degrees of freedom but in the same direction. It is then stated in [443] that

> Using non-proportional covariance matrices would give 4D-Var more freedom to explore various directions in which to fit the data. It is better to extend the space of directions being explored, even if the added directions are not optimal, than to constrain the algorithm in the direction already explored, and thereby introduce underdetermination along these directions.

An alternative approach presented and tested in [443] is to consider a tendency-based approximation. At ECMWF they run what is referred to as an **ensemble of 4D VARs**. We go into the specifics of this approach in Chapter 20, as the name suggests the center runs multiple 4D VARs in parallel, but each one of these analysis systems produces a set of initial conditions that can be used to initiate the global model. It is stated in [443] that the model forecasts produced from these different initial conditions represent a sample of the probability distribution for the true atmospheric state. The model tendencies derived from the ensemble members should represent a distribution of the possible evolution of the atmosphere from the true state. The different in tendencies can be interpreted as uncertainties in the model forcing or as an ensemble of possible realizations of the model error.

Once a model error covariance model has been decided upon, then the last feature of the weak constraint formulation is determining the length of the assimilation window. The restriction of the strong constraint 4D VAR is that the window length is limited by the range up to which model error can be neglected. Therefore in the weak constraint formulation we should be able to run with longer windows as we are damping the model error. At ECMWF they initially ran with 12-hour windows, but have now been able to move to running incremental 4D VAR with 24-hour windows.

Most operational numerical weather, and/or ocean, prediction centers employ some form of incremental VAR, if that is variational type of data assimilation they are using. Therefore we need to incrementalize (16.57). In [442], Trèmolet denotes the first guess by $\boldsymbol{x}_0^g$ for the initial conditions at the start of the assimilation window, but this state also linearizes the nonlinear forecast model as well as the observation operators, and define the increment to the control variable as $\delta\boldsymbol{x}_0 = \boldsymbol{x}_0 - \boldsymbol{x}_0^g$, such that the strong constraint incremental cost function and its Jacobian are as

$$J\left(\delta\boldsymbol{x}_0\right) = \frac{1}{2}\left(\delta\boldsymbol{x}_0 - \mathbf{b}\right)^T \mathbf{B}_0^{-1}\left(\delta\boldsymbol{x}_0 + \mathbf{b}\right) + \frac{1}{2}\sum_{i=1}^{N_o}\left(\mathbf{d}_i - \mathbf{H}_i\mathbf{M}_{0,i}\delta\boldsymbol{x}_0\right)^T \mathbf{R}_i^{-1}\left(\mathbf{d}_i - \mathbf{H}_i\mathbf{M}_{0,i}\delta\boldsymbol{x}_0\right), \tag{16.58a}$$

$$J_{\delta\boldsymbol{x}_0}\left(\delta\boldsymbol{x}_0\right) = \mathbf{B}_0^{-1}\left(\delta\boldsymbol{x}_0 + \mathbf{B}\right) - \sum_{i=0}^{N_0}\mathbf{M}_{0,i}^T\mathbf{H}_i^T\mathbf{R}_i^{-1}\left(\mathbf{d}_i + \mathbf{H}_i\mathbf{M}_{0,i}\delta\boldsymbol{x}_0\right) = \mathbf{0}, \tag{16.58b}$$

where $\mathbf{b} \equiv \boldsymbol{x}_0^g - \boldsymbol{x}_{b,0}$, $\mathbf{M}$ is the tangent linear model as defined earlier, and the innovation is defined by $\mathbf{d}_i = \boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}\left(\boldsymbol{x}_0^g\right)\right)$, respectively.

If we now introduce a first guess for the constant bias, denoted by $\boldsymbol{\beta}^g$, such that we can define the increment for the constant bias as $\delta\boldsymbol{\beta} = \boldsymbol{\beta}^t - \boldsymbol{\beta}^g$, and that we can define a cost function that is terms of the **augmented** variables, $\delta\boldsymbol{z} \equiv \begin{pmatrix} \delta\boldsymbol{x}_0 \\ \delta\boldsymbol{\beta} \end{pmatrix}$, and we can define a block diagonal matrix $\mathbf{A}$ such that $\begin{pmatrix} \mathbf{B}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{Q} \end{pmatrix}$, then we can write the full cost function as

$$J(\delta z_0) = \frac{1}{2}\left(\delta z + \widehat{\mathbf{b}}\right)^T \mathbf{A}^{-1}\left(\delta z + \widehat{\mathbf{b}}\right) + \frac{1}{2}\sum_{i=0}^{N_o}\left(\widehat{\mathbf{d}}_i - \widehat{\mathbf{H}}_i\widehat{\mathbf{M}}_{0,i}\delta z\right)^T \widehat{\mathbf{R}}_i^{-1}\left(\widehat{\mathbf{d}}_i - \widehat{\mathbf{H}}_i\widehat{\mathbf{M}}_{0,i}\delta z\right), \tag{16.59}$$

where $\widehat{\mathbf{b}} \equiv \begin{pmatrix} \mathbf{b} \\ \boldsymbol{\beta}^g \end{pmatrix}$, $\widehat{\mathbf{H}}_i \equiv \begin{pmatrix} \mathbf{H}_i & \mathbf{H}_i \end{pmatrix}$, $\widehat{\mathbf{M}}_{0,i} \equiv \begin{pmatrix} \mathbf{M}_{0,i} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$, $\widehat{\mathbf{R}}_i \equiv \begin{pmatrix} \mathbf{R}_i & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_i \end{pmatrix}$ and $\widehat{\mathbf{d}}_i \equiv \begin{pmatrix} \widetilde{\mathbf{d}}_i \\ \widetilde{\mathbf{d}}_i \end{pmatrix}$

where $\widetilde{\mathbf{d}}_i \equiv \boldsymbol{y}_i - \boldsymbol{h}_i\left(\mathcal{M}_{0,i}\left(\boldsymbol{x}_0^g + \boldsymbol{\beta}^g\right)\right)$.

Therefore the Jacobians of (16.59), with respect to $\delta\boldsymbol{x}_0$ and $\delta\boldsymbol{\beta}$, are

$$\nabla_{\delta z} J(\delta z) = \mathbf{A}\left(\delta z + \widehat{\mathbf{b}}\right) - \frac{1}{2}\sum_{i=0}^{N_o}\widehat{\mathbf{M}}_{0,i}^T\widehat{\mathbf{H}}_i^T\widehat{\mathbf{R}}_i^{-1}\left(\mathbf{d}_i - \widehat{\mathbf{H}}_i\widehat{\mathbf{M}}_{0,i}\delta z\right). \tag{16.60}$$

If we multiply out the matrix-vector products in (16.60), we see that we can separate out the two Jacobians, which in matrix equation form are:

$$\nabla J_{\delta x_0} = \mathbf{B}_0^{-1}\left(\delta\boldsymbol{x}_0 + \mathbf{b}\right) + \sum \mathbf{M}_{0,i}^T\mathbf{H}_i^T\mathbf{R}^{-1}\left(\mathbf{d}_i - \mathbf{H}_i\mathbf{M}_{0,i}\delta\boldsymbol{x}_0\right), \tag{16.61a}$$

$$\nabla J_{\delta\boldsymbol{\beta}} = \mathbf{Q}^{-1}\left(\delta\boldsymbol{\beta} + \boldsymbol{\beta}^g\right) + \sum \mathbf{H}_i^T\mathbf{R}^{-1}\left(\mathbf{d}_i - \mathbf{H}_i\mathbf{M}_{0,i}\delta\boldsymbol{x}_0\right). \tag{16.61b}$$

The important feature to note in the differences between (16.61a) and (16.61b) is that the model error component **does not** require the adjoint of the tangent linear model. An advantage of this formulation is that the tangent linear and adjoint models of the numerical model and the observation operators are untouched from the strong constraint formulation. In Fig. 16.8, we have plotted a simple illustration of four different sine waves where the red wave represents the background state, the blue wave represented the true state, the black wave represents the unbiased adjusted approach where the initial conditions have been optimized, and the green wave represents the state where we have analyzed initial conditions with a previous estimate of the bias added, but we are still $\delta\beta$ away from the true state.

16.5.3 Model Error Forcing Control Variable

In this situation the control vector contains the initial conditions and the model error $\boldsymbol{\eta}_i$. The problem now is how to define the model error. In [505] the model error was defined as $\nu_i = \lambda_i\Phi$, where $\boldsymbol{\Phi}$ is a three-dimensional field and the λ_is are predefined coefficients defining the evolution in time of the model error. In [503] the model error was defined as a first-order Markov variable in which the random variable could be defined on a coarser resolution in time and or space. In [161] it was proposed that the model error could be expressed through using a spectral representation of the form

$$\boldsymbol{\eta}_i = \boldsymbol{\gamma}_0 + \boldsymbol{\gamma}_1 \sin\frac{i\Delta t}{\tau} + \boldsymbol{\gamma}_2 \cos\frac{i\Delta t}{\tau},$$

where the coefficients $\boldsymbol{\gamma}_1$, $\boldsymbol{\gamma}_2$, and $\boldsymbol{\gamma}_2$ are three-dimensional fields, Δt is the time step, and finally τ is a time scale that the model error is thought to vary by.

In [442] Tremolet decides to break up the assimilation window into small interval upon which the model error is constant throughout that interval but can be different to the estimate of the model error in the next interval. These interval could be as long as the whole window which would be the bias scenario, or it could be every time step which would then be the full four-dimensional problem. Given this motivation, we can write the cost function for this scenario as

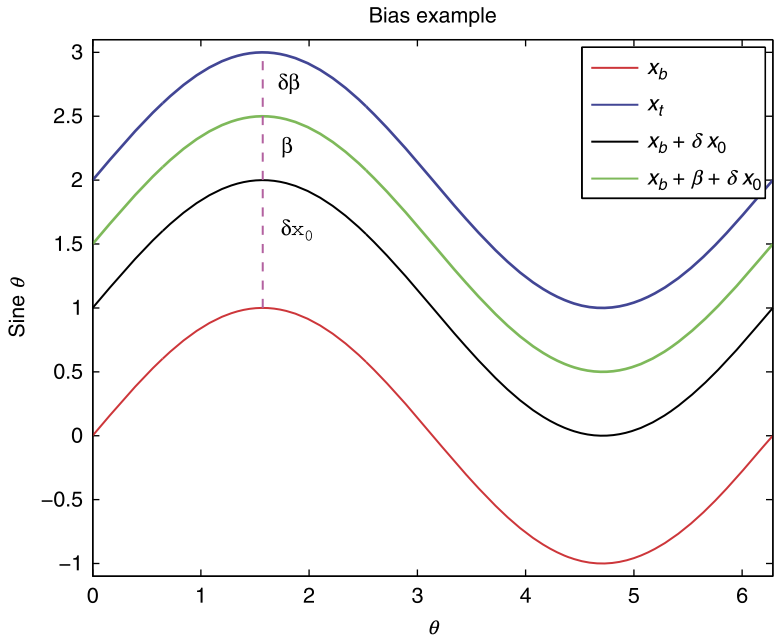


FIGURE 16.8

Simple illustration of the weak constraint constant bias parameter formulation.

$$J(\boldsymbol{x}_0, \boldsymbol{\eta}) = \frac{1}{2}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}_0^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_b\right) + \frac{1}{2}\sum_{i=1}^{N} \boldsymbol{\eta}_i^T \mathbf{Q}_i^{-1} \boldsymbol{\eta}_i + \frac{1}{2}\sum_{i=0}^{N}\left(\boldsymbol{y}_i - \boldsymbol{h}_i(\boldsymbol{x}_i)\right)^T \mathbf{R}_i^{-1}\left(\boldsymbol{y}_i - \boldsymbol{h}_i(\boldsymbol{x}_i)\right), \tag{16.62}$$

where to save on the notation we have that $\boldsymbol{x}_i \equiv \mathcal{M}_{0,i}(\boldsymbol{x}_0) + \boldsymbol{\eta}_i$ is the **forced model solution** [442].

The incremental formulation is obtained by introducing increments to $\boldsymbol{x}_0$ and $\boldsymbol{\eta}_i$, $\left(\delta\boldsymbol{x}_0, \delta\boldsymbol{\eta}_i\right)$, through the first guesses for the initial states and the model error as $\boldsymbol{x}_0^g$ and $\boldsymbol{\eta}_i^g$. However, for small perturbations $\delta\boldsymbol{x}_{i-1}$ and $\delta\boldsymbol{\eta}_i$ of $\boldsymbol{x}_{i-1}^g$ and $\boldsymbol{\eta}$, respectively, we are able to linearize the numerical model to describe the evolution of the perturbations as

$$\delta\boldsymbol{x}_i = \mathbf{M}_i \delta\boldsymbol{x}_{i-1} + \delta\boldsymbol{\eta}_i \equiv \mathbf{M}_{0,i}\delta\boldsymbol{x}_0 + \sum_{j}^{i} \mathbf{M}_{j,i}\delta\boldsymbol{\eta}_j. \tag{16.63}$$

The Jacobians of (16.63) with respect to $\delta\boldsymbol{x}_0$ and for the forcing term at a specific time t_i are given by

$$\nabla_{\delta\boldsymbol{x}_0} = \mathbf{B}_0^{-1}(\delta\boldsymbol{x}_0 - \mathbf{b}) + \sum_{j=0}^{N} \mathbf{M}_{0,j}^T \mathbf{H}_j^T \mathbf{R}_j^{-1}\left(\mathbf{d}_j + \mathbf{H}_j \mathbf{M}_{0,j}\delta\boldsymbol{x}_j\right), \tag{16.64a}$$

$$\nabla_{\delta\boldsymbol{\eta}_i} = \mathbf{Q}_0^{-1}\left(\delta\boldsymbol{\eta}_i - \boldsymbol{\eta}_i^g\right) - \sum_{j=1}^{N} \mathbf{M}_{i,j}^T \mathbf{H}_j^T \mathbf{R}_j^{-1}\left(\mathbf{d}_j + \mathbf{H}_j \mathbf{M}_{0,j}\delta\boldsymbol{x}_j\right). \tag{16.64b}$$

An important feature to note here, that is different to the constant bias formulation, is firstly; that the adjoint of the tangent linear forward model is present in (16.64b) and secondly that the increment, $\delta \boldsymbol{x}_j$, is a function of $\delta \boldsymbol{\eta}_j$ through (16.63). The main difference between this approach and the bias method is that the forcing directly modifies the model state in the forward integration, and as a consequence of this the adjoint appears in the Jacobian with respect to $\delta \boldsymbol{\eta}_j$.

16.5.4 Model State Control Variable

In this formulation of a model error approach the **four-dimensional** model state $\boldsymbol{x}$ is chosen as the control variable; this means that $\boldsymbol{x}$ is a function of time. Therefore the associated cost function is

$$J(\boldsymbol{x}) = \frac{1}{2}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}_0^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right) + \frac{1}{2}\sum_{i=0}^{N}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\boldsymbol{x}_i\right)\right)^T \mathbf{R}_i^{-1}\left(\boldsymbol{y}_i - \boldsymbol{h}_i\left(\boldsymbol{x}_i\right)\right) + \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{x}_i - \mathcal{M}_i\left(\boldsymbol{x}_{i-1}\right)\right)^T \mathbf{Q}_i^{-1}\left(\boldsymbol{x}_i - \mathcal{M}_i\left(\boldsymbol{x}_{i-1}\right)\right) \tag{16.65}$$

However, if we were to evaluate (16.65), then this would computationally expensive, so Trèmolet suggests breaking up the assimilation window into shorter intervals and as such the control variables are the initial conditions at the beginning at each of these intervals. Therefore we let m represent regularly spaced times in addition to the start of the window, and that the interval between these points each contain p time steps that start at the index $k_i = i \times p$ for $i = 1, 2, \ldots, m$. Thus (16.65) becomes

$$\begin{aligned} J(\boldsymbol{x}) = &\frac{1}{2}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right)^T \mathbf{B}_0^{-1}\left(\boldsymbol{x}_0 - \boldsymbol{x}_{b,0}\right) \\ &+ \frac{1}{2}\sum_{i=0}^{m}\sum_{j=0}^{p-1}\left(\boldsymbol{y}_{k_i+j} - \boldsymbol{h}_{k_i+j}\left(\mathcal{M}_{k_i}^{j}\boldsymbol{x}_{k_i}\right)\right)^T \mathbf{R}_{k_i+j}^{-1}\left(\boldsymbol{y}_{k_i+j} - \boldsymbol{h}_{k_i+j}\left(\mathcal{M}_{k_i}^{j}\boldsymbol{x}_{k_i+j}\right)\right) \\ &+ \frac{1}{2}\sum_{i=1}^{N}\left(\boldsymbol{x}_{k_i} - \mathcal{M}_{k_i}^{p}\left(\boldsymbol{x}_{k_{i-1}}\right)\right)^T \mathbf{Q}_{k_i}^{-1}\left(\boldsymbol{x}_i - \mathcal{M}_{k_i}^{p}\left(\boldsymbol{x}_{k_{i-1}}\right)\right), \end{aligned} \tag{16.66}$$

where $\mathcal{M}_i^j$ represents the nonlinear numerical model integrated for j steps to time t_i. If we were to set $m = 0$ and $p = n + 1$, then we would obtain the strong constraint version of 4D VAR.

We incrementalize (16.66) through setting the control variable as the correction $\delta\boldsymbol{x}$ to a guess $\boldsymbol{x}^g$ and if we now consider the gradient at the beginning of one of the sub-intervals at time set k_i, then we obtain

$$\begin{aligned} \nabla J_i = &\sum_{j=0}^{p-1}\left(\mathbf{M}_{k_i}^{j}\right)^T \mathbf{H}_{k_i+j}^T \mathbf{R}_{k_i+j}^{-1}\left(\mathbf{d}_{k_i+j} + \mathbf{H}_{k_i+j}\mathbf{M}_{k_i}^{p}\delta\boldsymbol{x}_{k_i}\right) \\ &+ \mathbf{Q}_{k_i}^{-1}\left(\delta\boldsymbol{x}_{k_i} - \mathbf{M}_{k_i}^{p}\delta\boldsymbol{x}_{k_{i-1}} + \mathbf{q}_{k_i}^{p}\right) - \left(\mathbf{M}_{k_{i+1}}^{P}\right)^T \mathbf{Q}_{k_{i+1}}^{-1}\left(\delta\boldsymbol{x}_{k_{i+1}} - \mathbf{M}_{k_{i+1}}^{p}\delta\boldsymbol{x}_{k_i} + \mathbf{q}_{k_{i+1}}\right), \end{aligned} \tag{16.67}$$

where $\mathbf{q}_{k_i}^{p} \equiv \boldsymbol{x}_{k_i}^{g} - \mathcal{M}_{k_i}^{p}\left(\boldsymbol{x}_{k_{i-1}}^{g}\right)$, and $\mathbf{M}_i^j$ represents the tangent linear model that is integrating for j steps starting at t_i.

16.5.5 Time Lag Model Error Modeling

Since the first edition of this textbook there has been some advancement on the implementation of the model error model at ECMWF for this operational data assimilation system, where in [43] there is an investigation into the structure of time-correlated model errors in this system.

In the opening paragraph of [43] there is a very telling remark about the state of data assimilation in 2021:

> While initial condition uncertainties can be effectively characterized through ensemble data assimilation and forecasting techniques, the treatment of systematic, time-correlated model errors has received considerably less attention, at least in the numerical weather prediction community, even though it increasingly appears to be one of the main obstacles to improved analysis and forecast accuracy and **reliability**.

We have emphasized the last word in this quote as it is our main motivator for working on improving data assimilation, even if the mathematics becomes a bit difficult. We strive to be able to provide the best mathematics to the operational centers so that they can provide the most reliable forecast, and as a result the public trust us when we say there is something bad coming, please take action.

Thus we now present the derivation of the measures from [43] to detect and attempt to compensate for time correlated model errors. We start with a system governed by linear dynamics where the evolution of the forecast error from one assimilation time ($t = n$) to the next ($t = n + 1$) can be written as:

$$\boldsymbol{\varepsilon}^f_{n+1} = \mathbf{M}\boldsymbol{\varepsilon}^a_n + \boldsymbol{\varepsilon}_n + \boldsymbol{\eta}_n, \tag{16.68}$$

where $\mathbf{M}$ is the prognostic model used to cycle the assimilation system, $\boldsymbol{\varepsilon}_n$ is an unbiased, $\mathbb{E}[\boldsymbol{\varepsilon}_n] = 0$, and time uncorrelated, $\mathbb{E}\left[\boldsymbol{\varepsilon}^q_{n+1}, \boldsymbol{\varepsilon}\right] = 0$ error that is completely characterized by its covariance matrix $\mathbf{Q_s} = \mathbb{E}[\boldsymbol{\varepsilon}_n, \boldsymbol{\varepsilon}_n] = 0$, where we have the additional model error term $\boldsymbol{\eta}_n$ that is allowed to be biased and serially correlated, but uncorrelated with the stochastic noise term $\boldsymbol{\varepsilon}^q_n$ and the observational errors.

A linear, unbiased assimilation update in the presence of a new batch of observations at time $t = n + 1$ will produce an analysis whose errors can be characterized as [43]:

$$\boldsymbol{\varepsilon}^a_{n+1} = \left(\mathbf{I} - \mathbf{K}_{n+1}\mathbf{H}\right)\boldsymbol{\varepsilon}^f_{n+1} + \mathbf{K}\boldsymbol{\varepsilon}^o_{n+1}, \tag{16.69}$$

where it analyzing the lagged time covariance of the analysis increments, that is:

$$\mathbf{A}^n_m \equiv \mathbb{E}\left[\left(x^a_m - x^f_m\right)\left(x^a_n - x^f_n\right)^T\right] = \mathbb{E}\left[\Delta x^a_m \left(\Delta x^a_n\right)^T\right]. \tag{16.70}$$

In [43] they are interested in the lag-1 covariance of the analysis increments for a data assimilation system, which implies $\mathbf{A}^a_{n+1} \equiv \mathbb{E}\left[\left(x^a_{n+1} - x^a_{n+1}\right)\left(x^a_n - x^f_n\right)^T\right]$. Through assuming statistical independence between observations and background and analysis errors at different update times, we can obtain:

$$\begin{aligned}
\mathbf{A}^n_{n+1} &= \mathbb{E}\left[\left(x^a_{n+1} - x^f_{n+1}\right)\left(x^a_n - x^f_n\right)^T\right], \\
&= \mathbb{E}\left[\mathbf{K}_{n+1}\left(\boldsymbol{\varepsilon}^o_{n+1} - \mathbf{H}\left(\mathbf{M}\boldsymbol{\varepsilon}^a_n + \boldsymbol{\varepsilon}^q_n + \boldsymbol{\eta}_n\right)\right)\left(\boldsymbol{\varepsilon}^a_n - \boldsymbol{\varepsilon}^f_n\right)\right],
\end{aligned}$$

$$= -\mathbf{K}_{n+1}\mathbf{H}\mathbf{M}\mathbb{E}\left[\boldsymbol{\varepsilon}_n^a\left(\boldsymbol{\varepsilon}_n^a - \boldsymbol{\varepsilon}_n^f\right)^T\right] - \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\varepsilon}_n^a - \boldsymbol{\varepsilon}_n^f\right)^T\right]. \tag{16.71}$$

The first term in (16.71) can be expanded as:

$$\begin{aligned}
-\mathbf{K}_{n+1}\mathbf{H}\mathbf{M}\mathbb{E}\left[\boldsymbol{\varepsilon}_n^a\left(\boldsymbol{\varepsilon}_n^a - \boldsymbol{\varepsilon}_n^f\right)^T\right] &= -\mathbf{K}_{n+1}\mathbf{H}\mathbf{M}\left(\mathbb{E}\left[\boldsymbol{\varepsilon}_n^a\left(\boldsymbol{\varepsilon}_n^a\right)^T\right]\right. \\
&\quad \left. -\left(\left(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n\right)\mathbb{E}\left[\boldsymbol{\varepsilon}_n^f\left(\boldsymbol{\varepsilon}_n^f\right)^T\right] + \mathbf{K}_{n+1}\mathbb{E}\left[\boldsymbol{\varepsilon}_n^o\left(\boldsymbol{\varepsilon}_n^f\right)^T\right]\right)\right), \\
&= -\mathbf{K}_{n+1}\mathbf{H}\mathbf{M}\left(\mathbf{P}_n^a - \left(\mathbf{I} - \mathbf{K}_n\mathbf{H}_n\right)\mathbf{P}_n^f\right).
\end{aligned} \tag{16.72}$$

In [43] they make the remark that the right-hand side of (16.72) vanishes in the case of a data assimilation system using an optimal gain matrix, a Kalman gain, that is, a matrix that minimizes the expected analysis error covariance; and states that this is in fact equivalent to the requirement that the residual analysis error be orthogonal to the analysis increment. In a data assimilation system where the model is not affected by a serially correlated model bias it is possible to recover a similar property of whiteness of the time series of analysis increments as that valid for the lagged innovation covariances.

In the case of time-correlated model error, however, the second term of (16.72) can still give a significant contribution to lag-1 analysis increment covariance. If we assume that this term is the dominant contribution, then through repeated applications of (16.68) and (16.69) we obtain:

The lag-1 analysis increment covariance is thus given by

$$\begin{aligned}
\mathbf{A}_{n+1}^n &\cong -\mathbf{K}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\varepsilon}_n^a - \boldsymbol{\varepsilon}_n^f\right)^T\right], \\
&= \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\varepsilon}_n^f\right)^T\right]\mathbf{H}^T\mathbf{K}_n^T, \\
&= \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\eta}_{n-1}\right)^T\right]\mathbf{H}^T\mathbf{K}_n^T + \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\varepsilon}_{n-1}^a\right)^T\right]\mathbf{M}^T\mathbf{H}^T\mathbf{K}_n^T, \\
&= \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\eta}_{n-1}\right)^T\right]\mathbf{H}\mathbf{K}_n^T + \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\eta}_{n-2}\right)^T\right]\left(\mathbf{I} - \mathbf{K}_{n-1}\mathbf{H}\right)^T\mathbf{M}\mathbf{H}^T\mathbf{K}_n^T \\
&+ \mathbf{K}_{n+2}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}\left(\boldsymbol{\varepsilon}_{n-2}^a\right)^T\right]\left(\mathbf{I} - \mathbf{K}_{n-1}\mathbf{H}\right)^T\mathbf{M}^T\mathbf{H}^T\mathbf{K}_n^T, \\
&= \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\eta}_{n-1}\right)^T\right]\mathbf{H}^T\mathbf{K}_n^T + \cdots \\
&+ \mathbf{K}_{n+1}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_n\left(\boldsymbol{\eta}_{n-k}\right)^T\right]\left(\left(\mathbf{I} - \mathbf{K}_{n-k+1}\mathbf{H}\right)^T\mathbf{M}^T\cdots\left(\mathbf{I} - \mathbf{K}_{n-1}\mathbf{H}\right)^T\mathbf{M}^T\right)\mathbf{H}^T\mathbf{K}_n^T + \cdots,
\end{aligned} \tag{16.73}$$

where $\mathbf{H}$ is the observation operator of the observing system, where here this is assumed to be stationary, and $\mathbf{K}_{n+1}$ is a generic gain matrix also valid at time $t = n + 1$.

In [43] they state that the lag-1 analysis increment covariance is thus given by the sum of a leading-order contribution proportional to the lag-1 model error covariance and higher-order contributions of the lagged model error covariances at increasing time separations. The higher-order contributions are expected to taper off rapidly, not only because of the decreasing covariances at increasing lag intervals, but also due to the repeated application of contraction operators in the analysis updates. The other important aspect is that the information obtained on the model error through the analysis increments is always mediated by the observing system (through the application of the observation operator H) and by the projection of this information back into model space through the Kalman gain matrix $\mathbf{K}$. In other

words, all the information we have on model error depends on the distribution and quality of available observations and the optimality of the analysis update. We should note here that the dependence on observation coverage may affect the reliability of the model error estimates and their diagnostics in localized regions, the spatially broad (∼1,000 km) model error covariances and the cycling strategy currently used in Weak Constraint WC-4DVar [235] at ECMWF act to greatly reduce the impact of local inhomogeneities in observation coverage.

The analysis increment lagged covariances for higher lag times have more complex analytical expressions according to [43], where they give an example of the lag-2 covariance expansion to leading terms as

$$\begin{aligned}\mathbf{A}^{n}_{n+2} &\equiv \mathbb{E}\left[\left(\mathbf{x}^{a}_{n+2}-\mathbf{x}^{f}_{n+2}\right)\left(\mathbf{x}^{a}_{n}-\mathbf{x}^{f}_{n}\right)^{T}\right],\\ &\cong \mathbf{K}_{n+2}\mathbf{H}\mathbb{E}\left[\boldsymbol{\eta}_{n+1}\left(\boldsymbol{\eta}_{n-1}\right)^{T}\right]\mathbf{H}^{T}\mathbf{K}_{n}^{T}+\mathbf{K}_{n+2}\mathbf{H}\mathbf{M}\left(\mathbf{I}-\mathbf{K}_{n+1}\mathbf{H}\right)\mathbb{E}\left[\boldsymbol{\eta}_{n}\left(\boldsymbol{\eta}_{n-1}\right]^{T}\right]\mathbf{H}^{T}\mathbf{K}_{n}^{T}\\ &+\cdots. \end{aligned} \tag{16.74}$$

Results using this measure, and a times series diagnostic can be found in [43], and the reader is referred to this paper for those findings.

16.6 Observational Errors

As we have seen in all of the versions of data assimilation presented so far, we assume that there are two type of errors associated with the observational component; measurement and representative errors, when combined are referred to as the **observational errors**. We also assumed that the observational errors are uncorrelated, which implies that the observational error covariance matrix, $\mathbf{R}$, is diagonal. With the introduction of satellites into operational data assimilation system for a variety of geophysical processes, the idea that the measurement errors associated with observations from different channels on a sensor are uncorrelated may not be valid; as such there has been research into defining correlations between channels on sensors, but also on quantifying how *filling in* the $\mathbf{R}$ matrix will affect the performance of the different operational data assimilation systems.

However, it is not just the measurements that are the cause of the observational errors. In numerical weather prediction, at least, there is a computational error as well due to numerical forecast models being ran at coarser scales than the some of the dynamical scales being observed. While this may not seem relevant to the observation component of the data assimilation system, we have to recall that we are not directly inverting the satellite observation to obtain estimates of the control variables, but in fact are transforming the model variables to try to match the observations through the possible nonlinear observation operator, $\mathbf{h}(\mathbf{x})$. We have to recognize that the observations are observing these smaller scales that the models are not resolving. This mismatch of the dynamical scales between the observations and the numerical model is referred to as **representative error**. There has been work recently to try to correctly formulate the probabilistic behavior of the representative error, and we refer the reader to the paper by Hodyss and Nichols [176] for the details of their approach.

In this section we shall introduce techniques to define and implement the correlated observational error into the variational data assimilation scheme at the United Kingdom's Met Office 4D VAR system.

16.6.1 Correlated Measurement Errors

In this section we shall introduce an approach that has been used to estimate the inter-channel observational error correlations for the Infrared Atmospheric Sounding Interferometer (IASI) with the Met Office 4D VAR [414]. In [414] the authors use a technique first presented in [95], which is a post-analysis diagnostic that utilizes the background and analysis departures. We shall briefly summarize the techniques from [95] before presenting some findings from [414].

The technique proposed in [95] is based upon linear estimation theory, similar to optimal interpolation, where the optimal analysis describing the true state of the atmosphere, for the case in [95], but is applicable to any geophysical situation, $\mathbf{x}^a$, can be expressed in terms of the background state, $\mathbf{x}^b$, and by what is referred to as the *background innovation*, denoted by $\mathbf{d}_b^o$ in [414], such that

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T \left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R}\right)^{-1} \mathbf{d}_b^o, \tag{16.75}$$

where $\mathbf{R}$ and $\mathbf{B}$ are the observational and background error covariance matrices seen throughout the data assimilation theory chapters, $\mathbf{H}$ is the Jacobian of the nonlinear observation operator $\mathbf{h}$. The background innovation vector is defined as the difference between the observation $\mathbf{y}$ and the associated background counterpart, $\mathbf{h}\left(\mathbf{x}^b\right)$, given by

$$\mathbf{d}_b^o = \mathbf{y} - \mathbf{h}\left(\mathbf{x}^b\right). \tag{16.76}$$

It is also possible to define the analysis innovation vector, denoted by $\mathbf{d}_a^o$, as

$$\mathbf{d}_a^o = \mathbf{y} - \mathbf{h}\left(\mathbf{x}^a\right). \tag{16.77}$$

Before we progress, we note that the background innovation can be expressed as

$$\mathbf{d}_b^o = \mathbf{y} - \mathbf{h}\left(\mathbf{x}^b\right) = \mathbf{y} - \mathbf{h}\left(\mathbf{x}^t\right) + \mathbf{h}\left(\mathbf{x}^t\right) - \mathbf{h}\left(\mathbf{x}^b\right) \approx \boldsymbol{\varepsilon}^{\boldsymbol{o}} - \mathbf{H}\boldsymbol{\varepsilon}_{\boldsymbol{b}}, \tag{16.78}$$

where $\boldsymbol{\varepsilon}^{\boldsymbol{o}}$ is the observational error and $\boldsymbol{\varepsilon}_{\boldsymbol{b}}$ is the background error. If we take the expectation of the square of (16.78), and assuming that the background and observational errors are uncorrelated, then we obtain

$$\mathbb{E}\left[\mathbf{d}_b^o \left(\mathbf{d}_b^o\right)^T\right] = \mathbb{E}\left[\boldsymbol{\varepsilon}^{\boldsymbol{o}} \left(\boldsymbol{\varepsilon}^{\boldsymbol{o}}\right)^T\right] + \mathbf{H}\mathbb{E}\left[\boldsymbol{\varepsilon}_{\boldsymbol{b}} \left(\boldsymbol{\varepsilon}_{\boldsymbol{b}}\right)^T\right]\mathbf{H} = \mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T. \tag{16.79}$$

The expression on the right-hand side in (16.79) that is important, as it says that if the observational error and the background error covariances in observations space are correct then we should satisfy (16.79).

In [95] the authors derive a series of diagnostics that can be used to determine consistency of the background, observational, and analysis errors. We are concerned with the observational consistency diagnostic here, which is given by

$$\begin{aligned}
\mathbf{d}_a^o &= \mathbf{y} - \mathbf{h}\left(\mathbf{x}^b + \delta\mathbf{x}^a\right) \approx \mathbf{y} - \mathbf{h}\left(\mathbf{x}^a\right) - \mathbf{H}\mathbf{K}\mathbf{d}_b^o, \\
&= \left(\mathbf{I} - \mathbf{H}\mathbf{K}\right)\mathbf{d}_b^o, \\
&= \mathbf{R}\left(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R}\right)^{-1}\mathbf{d}_b^o,
\end{aligned}$$

where $\mathbf{K}$ is the gain matrix that was defined in the optimal interpolation section.

We now take the expectation of the cross product of $\mathbf{d}_b^o$ and $\mathbf{d}_a^o$ such that

$$\mathbb{E}\left[\mathbf{d}_a^0\left(\mathbf{d}_b^o\right)^T\right] = \mathbf{R}\left(\mathbf{HBH}^T + \mathbf{R}\right)^{-1}\mathbb{E}\left[\mathbf{d}_b^o\left(\mathbf{d}_b^o\right)^T\right], \tag{16.80}$$

which simplifies to

$$\mathbb{E}\left[\mathbf{d}_a^o\left(\mathbf{d}_b^o\right)^T\right] = \mathbf{R}, \tag{16.81}$$

if the matrix $\mathbf{HK} = \mathbf{HBH}^T\left(\mathbf{HBH}^T + \mathbf{R}\right) - 1$ agree with exact covariances for the background and the observation error covariances.

Given the diagnostic equation in (16.81), Stewart et al. [414], mention Desroziers et al.'s comment that given the fact that we know that we are misrepresenting the observational error covariance structure. As such we should use the estimation technique as an iterative procedure; using the previously diagnosed $\mathbf{R}$ matrix at each iteration should result in a $\mathbf{R}$ matrix that is closer and closer to reality.

In [414] the authors apply this diagnostic and obtain the following plot which is a correlation plot for the 139 channels that were operationally assimilated in 2014 in Fig. 16.9, where we can clearly see that there are some relatively strong correlations between certain channels, which means that we are incorrectly approximating the observational error in the data assimilation schemes.

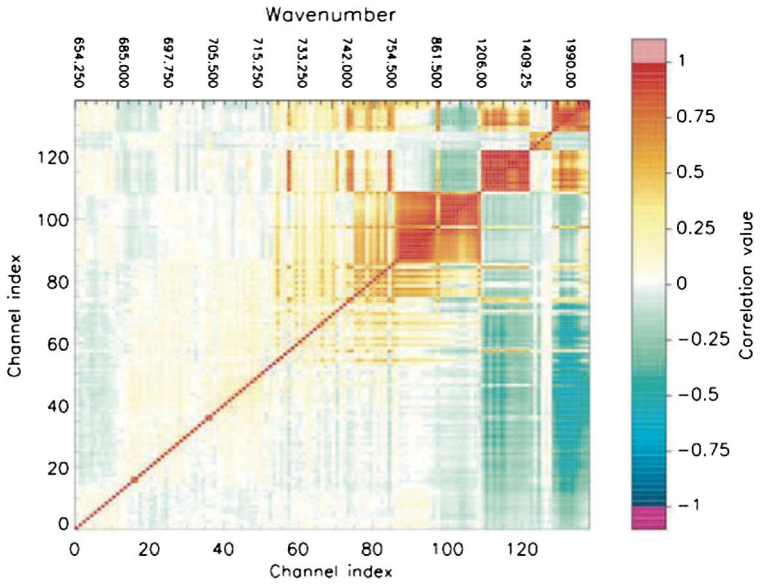


FIGURE 16.9

Copy of figure 8 of the correlated channels diagnostic from Stewart, L.M., Dance, S.L., Nichols, N.K., Eyre, J.R. and Cameron, J. (2014), Estimating interchannel observation-error correlations for IASI radiance data in the Met Office system. Q.J.R. Meteorol. Soc., 140: 1236-1244. https://doi.org/10.1002/qj.2211 https://creativecommons.org/licenses/by/3.0/.

There are some limitation of the usefulness of this diagnostic, but there has been some research recently to ascertain some form of uncertainty quantification for this measure [459]. We should note that most operational centers used to inflate the observational error variances on the diagonal of $\mathbf{R}$ to compensate for not modeling the correlations between the observations.

16.7 Forecast Sensitivity Observation Impact (FSOI)

We introduced Forecast Sensitivity of Observations (FSO) in the adjoint chapter, but now when we have a static 4D VAR system we can determine the FSO Impact (FSOI) on a forecast from a variational based data assimilation due to the adjoint and tangent linear models being available. In [213] they look at the impacts of different observations on the Unified Model at the Korean Meteorological Agency (KMA). The starting point to estimate the observation impact is to define the error of the forecast with respect to the true state measured by the moisture total energy norm, [267], as

$$\begin{aligned}\boldsymbol{\varepsilon} &= \left(\mathbf{x}^f - \mathbf{x}_t\right)^T \mathbf{C}\left(\mathbf{x}^f - \mathbf{x}_t\right), \\ &= \frac{1}{M_{domain}} \iiint \frac{1}{2}\left(\rho u'^2 + \rho v'^2 + \frac{\rho g^2}{\theta^2 N^2}\theta'^2 + \frac{1}{\rho c^2}P'^2 + \varepsilon\frac{\rho L^2}{C_p}q'^2\right) r^2 \cos\phi d\lambda d\phi dr, \end{aligned} \tag{16.82}$$

where $\mathbf{C}$ is a diagonal matrix that denotes the moisture energy norm, M_{domain} is the mass of the atmosphere in the model domain, N^2 is the square of the Brunt-Väisälä frequency, ρ is the air density, g and c are the gravitational acceleration and speed of sound respectively, r^2 is the square of the Earth's radius, ϕ and λ are the latitude and longitude coordinates, while u', v', θ', p', and q' represent the forecast errors of the zonal wind, meridional wind, potential temperature, pressure and specific humidity respectively. Finally ε is a factor used by [213] to control the forecast errors of the specific humidity. In the work presented in [213] the analysis from the data assimilation system is taken as the true state.

The nonlinear forecast error reduction (FER) is determined as the difference between the forecast errors with and without data assimilation. The difference between the total moisture energy of the forecast integrated from the analysis, $\left(\mathbf{x}_a^f - \mathbf{x}_t\right)^T \mathbf{C}\left(\mathbf{x}_a^f - \mathbf{x}_t\right)$ and that integrated from the background, $\left(\mathbf{x}_b^f - \mathbf{x}_t\right)^T \mathbf{C}\left(\mathbf{x}_b^f - \mathbf{x}_t\right)$ is expressed as, [213],

$$\begin{aligned}\delta e &= \left(\mathbf{x}_a^f - \mathbf{x}_t\right)^T \mathbf{C}\left(\mathbf{x}_a^f - \mathbf{x}_t\right) - \left(\mathbf{x}_b^f - \mathbf{x}_t\right)\mathbf{C}\left(\mathbf{x}_b^f - \mathbf{x}_t\right), \\ &= \left(\mathbf{x}_a^f - \mathbf{x}_b^f\right)^T \mathbf{C}\left(\left(\mathbf{x}_a^f - \mathbf{x}_t\right) + \left(\mathbf{x}_b^f - \mathbf{x}_t\right)\right). \end{aligned} \tag{16.83}$$

The analysis state, $\mathbf{x}_a$, is determined from the optimal linear analysis equation

$$\mathbf{x}_a = \mathbf{x}_b + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\mathbf{x}_b\right) = \mathbf{x}_b + \mathbf{K}\mathbf{d}, \tag{16.84}$$

where $\mathbf{K} = \mathbf{PH}\left(\mathbf{HPH}^T + \mathbf{R}\right)$. By applying the linearized model to (16.84) and using the relationships $\mathbf{x}_a^f \approx \mathbf{M}\mathbf{x}_a$ and $\mathbf{x}_b^f = \mathbf{M}\mathbf{x}_b$, δe is approximated in observation space as

$$\delta e \approx \mathbf{d}^T \mathbf{K}^T \mathbf{M}^T \mathbf{C}\left(\left(\mathbf{x}_a^f - \mathbf{x}_t\right) + \left(\mathbf{x}_b^f - \mathbf{x}_t\right)\right). \tag{16.85}$$

The approximated FER δe in observation space is referred to as the **observation impact**, and can be calculated independently for each observation assimilated.

The forecast sensitivity to observations is calculated as the gradient of δe with respect to $\mathbf{y}$,

$$\frac{\delta e}{\delta \mathbf{y}} \approx \mathbf{K}^T \mathbf{M}^T \mathbf{C} \left(\left(\mathbf{x}_a^f - \mathbf{x}_t \right) + \left(\mathbf{x}_b^f - \mathbf{x}_t \right) \right). \tag{16.86}$$

The observation impact corresponding to the ith observation type is expressed as

$$\delta e_i \approx \delta \mathbf{y}_i \left(\frac{\delta e}{\delta \mathbf{y}} \right)_i. \tag{16.87}$$

Here the adjoint of the perturbation forecast model is used in the observation impact estimation that is linearized about the forecast trajectory of the nonlinear model, here this is taken as the average between the $\mathbf{x}_a^f$ and $\mathbf{x}_b^f$. In the KMA UM the analyses are 3-h forecasts the adjoint of the perturbation forecast model is integrated 3 hours backwards from the analysis times for the observation impacts in (16.86) and (16.87). The reader is referred to [213] for more details of the study that they undertook, but we have a copy of their figure 2 in Fig. 16.10 which is a standard plot to determine the impact of different observations have on the metric that is being used to assess this.

16.8 Saddle Point 4D VAR

At the time of writing the first edition of this textbook, the idea of saddle point approximation for different configuration was being presented at conferences but was not yet published. However, there now have been advances in this field and so we now present a summary of this technique that comes form [124].

As stated in the abstract from [124], given the current evolution of computer architectures towards increasing parallelism, this requires a corresponding evolution towards more parallel data assimilation algorithms. The goal of this paper was to achieve parallelization in the time dimension of four-dimensional variational data assimilation.

The problem with 4D VAR in the formulation that has been presented is that it is highly sequential, where iterations of the minimization algorithm are performed sequentially, tangent linear and adjoint model integrations run one after the other. The way up until [124] for the parallelization of 4D VAR was achieved through parallelizing the tangent linear and adjoint models, using similar techniques that are used to parallelize the forecast model.

The approaches presented in [124] are for the weak constrain (model error) formulation of 4D VAR. It is assumed that the initial conditions and the model corrections provide sufficient information to perform a corrected integration of the model and therefore to determine a sequence of states throughout the analysis window. Thus, it is assumed that there exists a function, $\mathcal{F}$, that maps the initial state, $\mathbf{x}_0$, and a set of corrections, $\mathbf{q_1}, \mathbf{q_2}, \ldots, \mathbf{q_M}$, for $M > 0$ onto a sequence of states, $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N}$ for $N > 0$.

It is stated in [124] that the function $\mathcal{F}$ may not be invertible, but that they focus on the situation where it is with the weak-constraint 4D VAR system. Specifically, it is assumed that the model to be sufficiently accurate that short integrations, over subintervals of the full analysis window, can be used to regenerate intermediate states. The consequence of this is that it allows the four-dimensional state to

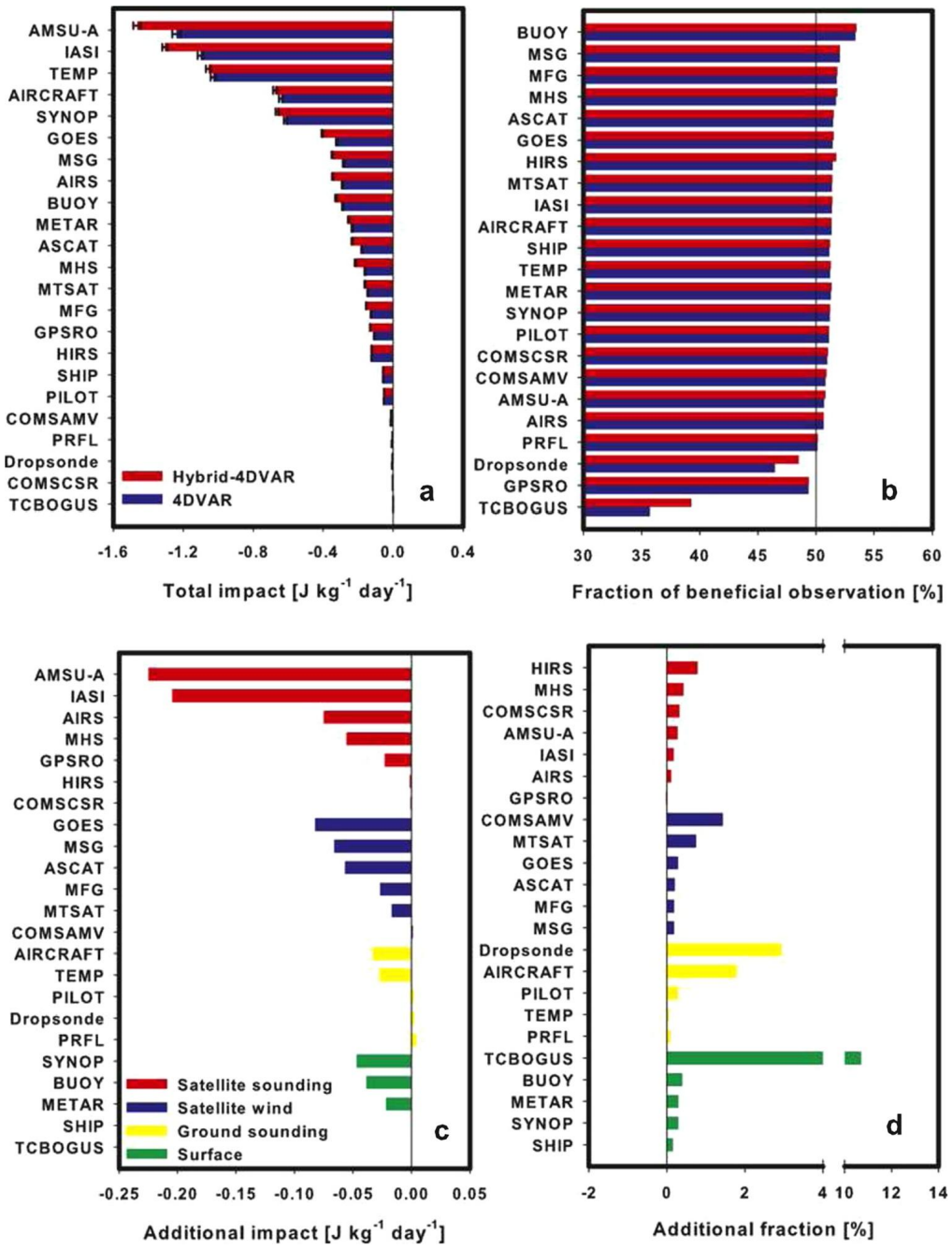


FIGURE 16.10

Copy of Figure 2 from Kim, S., & Kim, H. M. (2019). Forecast Sensitivity Observation Impact in the 4DVAR and Hybrid-4DVAR Data Assimilation Systems, Journal of Atmospheric and Oceanic Technology, 36(8), 1563-1575.

be characterized by the initial state $\mathbf{x_0}$, and a sequence of model-error corrections, $\mathbf{q_1}, \mathbf{q_2}, \ldots, \mathbf{q_N}$, that are applied at the end of each subinterval. In particular a set of states $\mathbf{x_0}, \ldots \mathbf{x_N}$ are considered that are valid at times $t_0, \ldots, t_N$ during an analysis window $t_0 \leq t \leq t_{N+1}$, and so we can define the function

$$\mathcal{F} : (\mathbf{x_0}; \mathbf{q_1}, \ldots, \mathbf{q_N}) \mapsto (\mathbf{x_0}, \mathbf{x_1}, \ldots, \mathbf{x_N}), \tag{16.88}$$

such that

$$\boldsymbol{x_k} = \mathcal{M}_k (\boldsymbol{x_{k-1}}) + \boldsymbol{q_k}, \tag{16.89}$$

for $k = 1, 2, \ldots, N$, where $\mathcal{M}_k$ denotes the operator that integrates the forecast model from time t_{k-1} to time t_k. Thus the associated cost function is given by

$$\begin{aligned} J(\underline{\mathbf{x}}) = &\frac{1}{2} (\mathbf{x_0} - \mathbf{x_b})^T \mathbf{B}^{-1} (\mathbf{x_0} - \mathbf{x_b}) + \frac{1}{2} \sum_{k=0}^{N} (\mathbf{y}_k - \mathbf{h}_k (\mathbf{x_k}))^T \mathbf{R}_k^{-1} (\mathbf{y}_k - \mathbf{h}_k (\mathbf{x_k})) \\ &+ \frac{1}{2} \sum_{k=1}^{N} (\mathbf{q_k} - \overline{\mathbf{q}})^T \mathbf{Q_k}^{-1} (\mathbf{q_k} - \overline{\mathbf{q}}), \end{aligned} \tag{16.90}$$

where we have introduced the new notation of $\overline{\mathbf{x}}$ to represent a four-dimensional quantity, we will use this notation again in Chapter 20, but is different to that used in [124]. Therefore,

$$\underline{\mathbf{x}} \equiv \begin{pmatrix} \mathbf{x_0} \\ \mathbf{x_1} \\ \vdots \\ \mathbf{x_N} \end{pmatrix}, \tag{16.91}$$

where the subscript k is used to identify the subintervals of the analysis window. The final term of the cost function in (16.90) penalizes deviations of the sequence $\mathbf{x_0}, \ldots, \mathbf{x_N}$ from a trajectory of the model. Note it assumed that the model errors are Gaussian distributed with mean $\overline{\mathbf{q}}$, and covariance

$$\mathbb{E}\left[(\mathbf{q_i} - \overline{\mathbf{q}}) (\mathbf{q_j} - \overline{\mathbf{q}})^T \right] = \begin{cases} \mathbf{Q_i} & i = j, \\ \mathbf{0} & i \neq j. \end{cases} \tag{16.92}$$

In [124] they now introduce two names for different configurations of (16.90). The first is referred to as the *4D-state formulation*, where they use (16.89) to eliminate the $\mathbf{q_k}$s so that the cost functions is in terms of $\underline{\mathbf{x}}$. The second formulation is referred to as the *forcing formulation* where it is the $\mathbf{x_k}$s that are eliminated.

As we mentioned earlier, most operational numerical weather prediction centers use a form of incremental VAR. Thus the next step is to linearize (16.90) about the sequence of states $\mathbf{x_0}^{(n)}, \mathbf{x_1}^{(n)}, \ldots \mathbf{x_N}^{(n)}$, where the superscript is referred to as the outer iteration index, referring to the outer loop. Thus the incremental cost function is given by

$$\begin{aligned} J^{(n)} = &\frac{1}{2} \left(\delta \mathbf{x_0} - \mathbf{b}^{(n)} \right)^T \mathbf{B}^{-1} \left(\delta \mathbf{x_0} - \mathbf{b}^{(n)} \right) \\ &+ \frac{1}{2} \sum_{k=0}^{N} \left(\mathbf{d}_k^{(n)} - \mathbf{H}^{(n)} \delta \mathbf{x}_k \right)^T \mathbf{R}_k^{-1} \left(\mathbf{d}_k^{(n)} - \mathbf{H}^{(n)} \delta \mathbf{x}_k \right) \left(\mathbf{d}_k^{(n)} - \mathbf{H}^{(n)} \delta \mathbf{x}_k \right) \end{aligned}$$

$$+\frac{1}{2}\sum_{k=1}^{N}\left(\delta\mathbf{q_k}-\mathbf{c_k}^{(n)}\right)^T\mathbf{Q}_k^{-1}\left(\delta\mathbf{q_k}-\mathbf{c_k}^{(n)}\right), \tag{16.93}$$

where

$$\mathbf{b}^{(n)}\equiv\mathbf{x}_b-\mathbf{x_0^{(n)}}, \tag{16.94}$$

$$\mathbf{c}^{(n)}\equiv\overline{\mathbf{q}}-\mathbf{q_k^{(n)}}, \tag{16.95}$$

$$\mathbf{d}^{(n)}\equiv\mathbf{y}_k-\mathbf{h}_k\left(\mathbf{x_k^{(n)}}\right), \tag{16.96}$$

and $\mathbf{h_k^{(n)}}$ is the tangent linear of the observation operator about the nth-iteration nonlinear state, with the model-error correction increment $\delta\mathbf{q_k}$ is defined via the linearized version of (16.89);

$$\delta\mathbf{x_k}=\mathbf{M}_k^{(n)}\delta\mathbf{x_{k-1}}+\delta\mathbf{q}_k, \tag{16.97}$$

and $\mathbf{M}_k^{(n)}$ denotes the linearization of $\mathcal{M}_k$ about a trajectory of the nonlinear model with initial conditions $\mathbf{x_{k-1}}^{(n)}$.

Minimization of the quadratic cost function above results in a set of increments, $\delta\mathbf{x_k^{(n)}}$, to the states $\mathbf{x_0^{(n)}}, \mathbf{x_1^{(n)}}, \ldots, \mathbf{x_N^{(n)}}$. One possible incremental algorithm that is presented in [124] is to define the linearization states for the $(n+1)$th outer iteration as

$$\mathbf{x_k^{(n+1)}}=\mathbf{x_k^{(n)}}+\delta\mathbf{x_k^{(n)}}, \tag{16.98}$$

for $k=0,\ldots,N$. Once this state has been calculated for each k it is necessary to update the quantities in (16.94)–(16.96), along with the linearizations of the nonlinear model, and the observation operator.

The alternative incremental algorithm may be derived by first calculating $\delta\mathbf{q}_k$ using $\delta\mathbf{q_k}=\delta\mathbf{x_k}-\mathbf{M}_k^{(n)}\delta\mathbf{x_{k-1}}$, and then using these increments to update the model error corrections at the outer loop:

$$\mathbf{q_k^{(n+1)}}=\mathbf{q_k^{(n)}}+\delta\mathbf{q_k^{(n)}}, \tag{16.99}$$

for $k=1,\ldots,N$. The initial state must also be updated:

$$\mathbf{x_0^{(n+1)}}=\mathbf{x_0^{(n)}}+\delta\mathbf{x_0^{(n)}}. \tag{16.100}$$

We now consider the inner loop which we will reformulate in a way to introduce the saddle point approach to help with the parallelization. We start by introducing vectors:

$$\delta\underline{\mathbf{x}}\equiv\begin{pmatrix}\delta\mathbf{x_0}\\ \delta\mathbf{x_1}\\ \vdots\\ \delta\mathbf{x_N}\end{pmatrix},\qquad \delta\underline{\mathbf{p}}\equiv\begin{pmatrix}\delta\mathbf{x_0}\\ \delta\mathbf{q_1}\\ \vdots\\ \delta\mathbf{q_N}\end{pmatrix}. \tag{16.101}$$

We now introduce the following four-dimensional arrays:

$$\underline{\mathbf{R}} \equiv \begin{pmatrix} \mathbf{R_0} & & & \\ & \mathbf{R_1} & & \\ & & \ddots & \\ & & & \mathbf{R_N} \end{pmatrix},$$

$$\underline{\mathbf{D}} \equiv \begin{pmatrix} \mathbf{B} & & & \\ & \mathbf{Q_1} & & \\ & & \ddots & \\ & & & \mathbf{Q_N} \end{pmatrix},$$

$$\underline{\mathbf{H}} \equiv \begin{pmatrix} \mathbf{H_0} & & & \\ & \mathbf{H_1} & & \\ & & \ddots & \\ & & & \mathbf{H_N} \end{pmatrix},$$

with the two four-dimensional vectors:

$$\underline{\mathbf{d}} \equiv \begin{pmatrix} \mathbf{d_0} \\ \mathbf{d_1} \\ \vdots \\ \mathbf{d_N} \end{pmatrix}, \qquad \underline{\mathbf{b}} \equiv \begin{pmatrix} \mathbf{b} \\ \mathbf{c_1} \\ \vdots \\ \mathbf{c_N} \end{pmatrix},$$

that enables the inner loop quadratic cost function to be written as:

$$J = \frac{1}{2}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right)^T \underline{\mathbf{D}}^{-1}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right) + \frac{1}{2}\left(\underline{\mathbf{d}} - \underline{\mathbf{H}}\delta\underline{\mathbf{x}}\right)^T \underline{\mathbf{R}}^{-1}\left(\underline{\mathbf{d}} - \underline{\mathbf{H}}\delta\underline{\mathbf{x}}\right). \tag{16.102}$$

The next step is to recall that we mentioned the function, $\mathcal{F}$, where in this inner loop this corresponds to a matrix $\underline{\mathbf{F}}$ that maps $\delta\underline{\mathbf{p}}$ on to $\delta\underline{\mathbf{x}}$ as

$$\delta\underline{\mathbf{x}} = \underline{\mathbf{F}}\delta\underline{\mathbf{p}} \tag{16.103}$$

Through using $\delta\mathbf{q_k} = \delta\mathbf{x_k} - \mathbf{M_k^{(n)}}\delta\mathbf{x_{k-1}}$, $\mathbf{F}$ is invertible such that

$$\underline{\mathbf{F}}^{-1} \equiv \begin{pmatrix} \mathbf{I} & & & & \\ -\mathbf{M_1} & \mathbf{I} & & & \\ & -\mathbf{M_2} & \mathbf{I} & & \\ & & \ddots & \ddots & \\ & & & -\mathbf{M_N} & \mathbf{I} \end{pmatrix}. \tag{16.104}$$

It is then stated in [124] that an explicit matrix representation of $\underline{\mathbf{F}}$ can be written as

$$\underline{\mathbf{F}} \equiv \begin{pmatrix} \mathbf{I} & & & & \\ \mathbf{M_{1,1}} & \mathbf{I} & & & \\ \mathbf{M}_{1,2} & \mathbf{M_{2,2}} & \mathbf{I} & & \\ \vdots & \vdots & \ddots & \ddots & \\ \mathbf{M_{1,N}} & \mathbf{M_{2,N}} & \cdots & \mathbf{M_{NN}} & \mathbf{I} \end{pmatrix}, \tag{16.105}$$

where $\mathbf{M_{i,j}} = \mathbf{M_j} \ldots \mathbf{M_i}$, represents an integration of the linear model from time t_{i-1} to t_j.

However, it is stated in [124] that the integration of the linear model that are required to calculate $\delta\underline{\mathbf{p}}$ from $\delta\underline{\mathbf{x}}$ can, in principle, be performed in parallel, since $\delta\underline{\mathbf{x}}$ contains the initial states for the model integrations $\mathbf{M_1}, \mathbf{M_2}, \ldots, \mathbf{M_N}$. However, the calculation of $\delta\underline{\mathbf{x}}$ from $\delta\underline{\mathbf{p}}$ requires a sequence of integrations of the linear model to determine first $\delta\underline{\mathbf{x}_1}$, then $\delta\underline{\mathbf{x}_2}$, then $\delta\underline{\mathbf{x}_3}$, ...

To overcome this the saddle approach is introduced in [124] where now we have a new variable, $\delta\underline{\mathbf{w}} = \underline{\mathbf{H}}\delta\underline{\mathbf{x}}$, and rewrite (16.102) as

$$J = \frac{1}{2}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right)^T \underline{\mathbf{D}}^{-1}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right) + \frac{1}{2}\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{d}}\right)^T \underline{\mathbf{R}}^{-1}\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{d}}\right). \tag{16.106}$$

In this formulation, the cost function is an explicit function of both $\delta\underline{\mathbf{p}}$ and $\delta\underline{\mathbf{w}}$ and is implicitly also a function of $\delta\underline{\mathbf{x}}$ via the equations $\delta\underline{\mathbf{p}} = \underline{\mathbf{F}}^{-1}\delta\underline{\mathbf{x}}$ and $\delta\underline{\mathbf{w}} = \underline{\mathbf{H}}\delta\underline{\mathbf{x}}$.

As a constrained minimization problem, Lagrange multipliers $\underline{\boldsymbol{\lambda}}$ and $\underline{\boldsymbol{\mu}}$ which defines the Lagrangian:

$$\begin{aligned} \mathcal{L} = &\frac{1}{2}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right)^T \underline{\mathbf{D}}^{-1}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right) + \frac{1}{2}\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{d}}\right)^T \underline{\mathbf{R}}^{-1}\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{d}}\right) \\ &+ \underline{\boldsymbol{\lambda}}^T\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{F}}^{-1}\delta\underline{\mathbf{x}}\right) + \underline{\boldsymbol{\mu}}^T\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{H}}\delta\underline{\mathbf{x}}\right). \end{aligned} \tag{16.107}$$

The minimum of the cost function coincides with the stationary point of $\mathcal{L}$, which may be found by setting its derivatives with respect to $\delta\underline{\mathbf{p}}$, $\delta\underline{\mathbf{w}}$, $\delta\underline{\mathbf{x}}$, $\underline{\boldsymbol{\lambda}}$, and $\underline{\boldsymbol{\mu}}$ to zero, which results in

$$\underline{\mathbf{D}}^{-1}\left(\delta\underline{\mathbf{p}} - \underline{\mathbf{b}}\right) + \underline{\boldsymbol{\lambda}} = \underline{\mathbf{0}}, \tag{16.108}$$

$$\underline{\mathbf{R}}^{-1}\left(\delta\underline{\mathbf{w}} - \underline{\mathbf{d}}\right) + \underline{\boldsymbol{\mu}} = \underline{\mathbf{0}}, \tag{16.109}$$

$$\left(\underline{\mathbf{F}}^{-1}\right)^T \underline{\boldsymbol{\lambda}} + \underline{\mathbf{H}}^T\underline{\boldsymbol{\mu}} = \underline{\mathbf{0}}, \tag{16.110}$$

$$\delta\underline{\mathbf{p}} - \underline{\mathbf{F}}^{-1}\delta\underline{\mathbf{x}} = \underline{\mathbf{0}}, \tag{16.111}$$

$$\delta\underline{\mathbf{w}} - \underline{\mathbf{H}}\delta\underline{\mathbf{x}} = \underline{\mathbf{0}}. \tag{16.112}$$

Eliminating $\delta\underline{\mathbf{p}}$ and $\delta\underline{\mathbf{w}}$ from (16.108) and (16.109) using the constraint equations, (16.111) and (16.112), and after a slight rearrangement results in the following single matrix equation:

$$\mathcal{A}\begin{pmatrix} \underline{\boldsymbol{\lambda}} \\ \underline{\boldsymbol{\mu}} \\ \delta\underline{\mathbf{x}} \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{b}} \\ \underline{\mathbf{d}} \\ \underline{\mathbf{0}} \end{pmatrix}, \tag{16.113}$$

where

$$\mathcal{A} \equiv \begin{pmatrix} \underline{\mathbf{D}} & \underline{\mathbf{0}} & \underline{\mathbf{F}}^{-1} \\ \underline{\mathbf{0}} & \underline{\mathbf{R}} & \underline{\mathbf{H}} \\ \left(\underline{\mathbf{F}}^{-1}\right)^T & \underline{\mathbf{H}}^T & \underline{\mathbf{0}} \end{pmatrix}. \tag{16.114}$$

The equation in (16.113) is referred to as a **saddle-point equation** due to the shape of the Lagrangian $\mathcal{L}$, that has positive and negative curvature. An advantage of this formulation is that is allows for a further level of parallelism due to multiplication of a vector by $\mathcal{A}$ requires evaluation of the quantities: $\underline{\mathbf{D}}\underline{\boldsymbol{\lambda}}$, $\underline{\mathbf{F}}^{-1}\delta\underline{\mathbf{x}}$, $\underline{\mathbf{R}}\underline{\boldsymbol{\mu}}$, $\left(\underline{\mathbf{F}}^{-1}\right)^T\underline{\boldsymbol{\lambda}}$, and $\underline{\mathbf{H}}^T\underline{\boldsymbol{\mu}}$. It is stated in [124] that all of these calculations can be performed simultaneously, since the vector $\underline{\boldsymbol{\lambda}}$, $\underline{\boldsymbol{\mu}}$ and $\delta\underline{\mathbf{x}}$ are known at the start of the calculation. A particular advantage of this approach is the ability to evaluate $\underline{\mathbf{F}}^{-1}\delta\underline{\mathbf{x}}$ and $\left(\underline{\mathbf{F}}^{-1}\right)^T\underline{\boldsymbol{\lambda}}$ at the same time means that the integrations of the tangent-linear and adjoint models may be conducted simultaneously.

The work in [124] is for the weak constraint version of 4D VAR, but it is the extension of the work on strong constraint 4D VAR as a saddle point problem in [232]. While we do not go into details about how to solve this problem here, we will introduce a pre-conditioner that has been derived for a formulation that is similar to this in the next chapter. However, the reader is referred to [124] for more of the algorithmic details.

16.9 Rapid Update Cycling (RUC)

The theory that we summarize here is from [330], and we start with the caveat that the author has in his paper in that we are not referring to Rapid Update Cycle, also referred to as RUC which is the current, at time of writing, National Centers for Environmental Prediction's operational global forecast-analysis system over North America

The opening paragraphs of [330] lay the foundations of what will be referred to Rapid Update Cycling, but more importantly with **overlapping windows**, by stating that in the traditional cycling of forecasts and data assimilation for numerical weather prediction, the DA step for the global model has occurred every 6 or 12 hours. This was appropriate for an era when data was concentrated at the main synoptic times, and the limited area models (LAMs) for which the global model provides boundary conditions were cycled every 6 hours. However, in recent years data has become dominated by sources that are essentially continuous in time, and centers such as the Met Office will soon cycle their highest resolution LAM every hour.

By increasing the frequency of global analyses, for example to every hour, global forecasts can be based on more recent data, which is not only desirable in itself, but provides timely lateral boundary conditions (LBCs) for high resolution LAMs. Furthermore, by having more frequent analyses the analysis increments will be smaller, which will improve the validity of the linear approximations in DA schemes. More frequent analyses may also improve the affordability of DA methods as the computational load is distributed more evenly in time.

As a result of the delay in receiving some data, to ensure that all the data that are received are also assimilated, the assimilation windows will need to **overlap**. The optimal solution to this problem is achieved but involves manipulating simultaneously all the states in the window and their joint errors.

The other motivation for the work in [330] is due to fact that observations are not received instantaneously. As an example, [330] explains that; by 09Z on 18 June 2015 the Met Office had received over 80 million observations, valid between 09Z on 15 June and 03Z on 18 June 2015, including around 0.8 million surface, 2.1 million aircraft and sonde, 12.4 million satwind and 14 million ATOVS observations. The delay between validity time and receipt for these observation types is recorded in Fig. 16.11, which is a copy of figure 1 from [330], where it is clear that to receive 95% of aircraft and sonde, surface, ATOVS and satwind observation took respectively 0.6, 1.5, 3.4 and 4.1 hours.

As you can imagine this presents a quandary for traditional cycling that aims to produce an analysis every 6, 12, or 24 hours. For example, we consider a 4D VAR system with a 6-hour window, $[T-3, T+3]$, that generates an analysis at $T-3$, where the units here are hours. It is stated in [330] that we could perform the analysis at $T+3$ using all observations available by $T+3$, which would minimize the time delay to produce the analysis, but observations received after $T+3$ would not be assimilated. Alternatively, one could perform the analysis at T+7, by which time, Fig. 16.11, almost all the observations valid in the window have been received, but the analysis is only available 4 hours after the end of the window and 10 hours after its beginning. To generate an estimate of state at $T+7$ it possible we could run a 10-hour forecast from the analysis, but compared with the estimate of state at $T-3$ this will be **degraded by model error**.

[330] states that centers such as the Met Office mitigate these issues by performing each analysis twice, a *late cut-off* analysis at about $T+6$, and an *early cut-off* analysis at about $T+3$. [330] provides a illustration of this setup, and we have a copy of this diagram in Fig. 16.12, where this figure shows two adjacent, non-overlapping, windows $[3Z, 9Z)$ and $[9Z, 15Z)$. As an example from [330], at $8Z$ valid observations are received between $4Z$ and $8Z$. If we now consider the window $[3Z, 9Z)$, then for the early cut-off run where the data assimilation is performed at $9Z$, and use the observations in

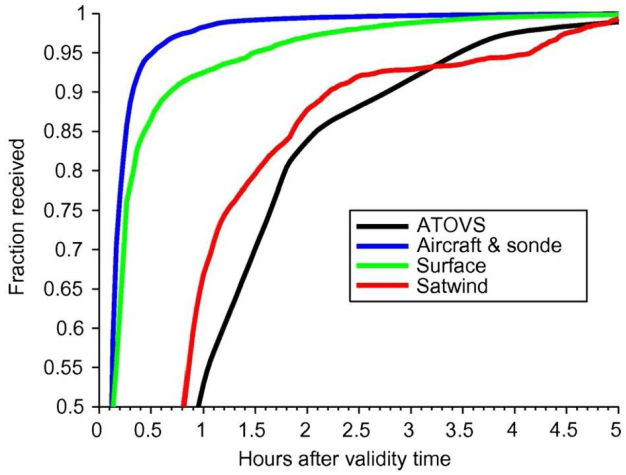


FIGURE 16.11

Copy of figure 1 from T. J. Payne (2017) Rapid update cycling with delayed observations, Tellus A: Dynamic Meteorology and Oceanography, 69:1, DOI: 10.1080/16000870.2017.1409061. https://creativecommons.org/licenses/by/4.0/.

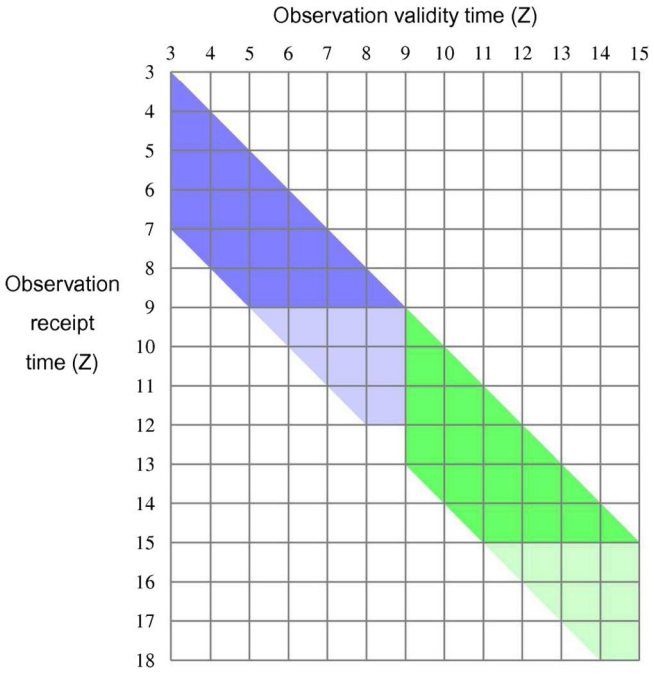


FIGURE 16.12

Copy of figure 2 from T. J. Payne (2017) Rapid update cycling with delayed observations, Tellus A: Dynamic Meteorology and Oceanography, 69:1, DOI: 10.1080/16000870.2017.1409061. https://creativecommons.org/licenses/by/4.0/.

the dark blue region (Fig. 16.12), and for the late cut-off run perform it at approximately 12Z and use virtually all observations ever valid in the window (combined light and dark blue region) in Fig. 16.12.

In [330] it is stated that having both early and late cut-off analyses goes some way to mitigating the shortcomings of 6-hourly cycling. However, the analyses are still 6 hours apart, which makes them insufficiently timely for some purposes, notably the LBCs for hourly LAM analyses; the analysis increment is much larger than would be the case with an hourly update, so nonlinearity can be a significant problem, especially for the linear model in 4D VAR; and the approach is inefficient insofar as the early cut-off analyses are not used as part of a cycle.

We have a copy of figure 3 from [330] in Fig. 16.13, that serves to illustrate how we would like to deal with the same case: each hour we assimilate all observations received in the last hour, for example at 12Z we assimilate the observations received between 11Z and 12Z (green region); these are valid between 7Z and 12Z. In principle we do not re-assimilate the observations valid between 7Z and 12Z received at earlier times (blue, red yellow and cyan regions) as the information from these observations has been **transferred to previous analyses and thereby to the background for this cycle**. In the context of the work from [330] they refer to any cycling where observations are assimilated as soon as they are received, or within some short time of receipt, as **rapid update cycling (RUC)**.

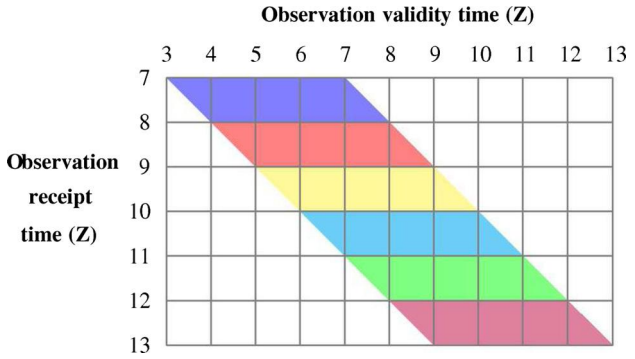


FIGURE 16.13

Copy of figure 3 from T. J. Payne (2017) Rapid update cycling with delayed observations, Tellus A: Dynamic Meteorology and Oceanography, 69:1, DOI: 10.1080/16000870.2017.1409061. https://creativecommons.org/licenses/by/4.0/.

The formal mathematics to describe this situation start from the assumption that the observations are valid at exact multiples of time increments δt, note this is different from the time step of the numerical model Δt, and become available after *delays* of $0, \delta t, 2\delta t, \ldots, N\delta t$. In [330] the following notation is introduced where the superscripts denote when the observations are received, while the subscripts represents their validity time, where the longest delay being $N\delta t$:

$$\mathbf{y}_{\mathbf{k}}^{(\mathbf{k})}, \mathbf{y}_{\mathbf{k-1}}^{(\mathbf{k})}, \mathbf{y}_{\mathbf{k-2}}^{(\mathbf{k})}, \cdots, \mathbf{y}_{\mathbf{k-N}}^{(\mathbf{k})}. \tag{16.115}$$

The next step in [330] is to introduce notation for the concatenation of $N+1$ vectors, which is the underline feature introduce in the last chapter, while [330] is not seeking a saddle point solution here, but seeking a form of 4D analysis.

The first different feature in [330] is to define $\underline{\mathbf{y}}_k$ to be the observations received at $k\delta t$ as

$$\underline{\mathbf{y}}_k \equiv \begin{pmatrix} \mathbf{y}_{\mathbf{k-N}}^{(\mathbf{k})} \\ \vdots \\ \mathbf{y}_{\mathbf{k}}^{(\mathbf{k})} \end{pmatrix}. \tag{16.116}$$

The next step is to seek the expectation of $\underline{\mathbf{x}}_k$ and $\underline{\mathbf{x}}_{k+1}$, given observations received up to time $k\delta t$, as

$$\mathbb{E}\left[\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \underline{\mathbf{y}}_1, \ldots, \underline{\mathbf{y}}_k\right], \mathbb{E}\left[\underline{\mathbf{x}}_{k+1} \middle| \underline{\mathbf{y}}_0, \underline{\mathbf{y}}_1, \ldots, \underline{\mathbf{y}}_k\right], \tag{16.117}$$

where

$$\mathbf{y}_i^{(k)} = \mathbf{h}_i^{(k)}(\mathbf{x}_i) + \boldsymbol{\nu}_i^{(k)}, \tag{16.118}$$

with model here denoted as $\mathbf{f}_i$ for each i as

$$\mathbf{x}_{i+1} = \mathbf{f}_i^{(k)}(\mathbf{x}_i) + \boldsymbol{\omega}_i, \tag{16.119}$$

where $\boldsymbol{\nu}_i^{(k)}$ and $\boldsymbol{\omega}_i$ are errors of known distributions.

It is now stated that given $\underline{\mathbf{x}}_k$, and assuming $\boldsymbol{\omega}_K$ has zero mean, the best fit estimate of $\underline{\mathbf{x}}_{k+1}$ before observations received at $k\delta t$ are assimilated, is

$$\underline{\mathbf{f}}_k\left(\underline{\mathbf{x}}_k\right) \equiv \begin{pmatrix} \mathbf{x}_{k-N-1} \\ \vdots \\ \mathbf{x}_k \\ \mathbf{f}_k\left(\mathbf{x}_k\right) \end{pmatrix}. \tag{16.120}$$

We now define

$$\underline{\boldsymbol{\omega}} \equiv \begin{pmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \boldsymbol{\omega}_k \end{pmatrix}, \qquad \underline{\boldsymbol{\nu}}_k \equiv \begin{pmatrix} \boldsymbol{\nu}_{k-N}^{(k)} \\ \vdots \\ \boldsymbol{\nu}_{k-1}^{(k)} \\ \boldsymbol{\nu}_k^{(k)} \end{pmatrix}, \qquad \underline{\mathbf{h}}_k\left(\underline{\mathbf{x}}_k\right) \equiv \begin{pmatrix} \mathbf{h}_{k-N}^{(k)}\left(\mathbf{x}_{k-N}\right) \\ \vdots \\ \mathbf{h}_{k-1}^{(k)}\left(\mathbf{x}_{k-1}\right) \\ \mathbf{h}_k^{(k)}\left(\mathbf{x}_k\right) \end{pmatrix}. \tag{16.121}$$

This then enables us to write (16.118) and (16.119) respectively as

$$\underline{\mathbf{y}}_k = \underline{\mathbf{h}}_k\left(\underline{\mathbf{x}}_k\right) + \underline{\boldsymbol{\nu}}_k, \tag{16.122}$$

$$\underline{\mathbf{x}}_{k+1} = \underline{\mathbf{f}}_k\left(\underline{\mathbf{x}}_k\right) + \underline{\boldsymbol{\omega}}_k, \tag{16.123}$$

where it is stated in [330] that these equations are in the standard form for observation and signal map equations in estimation theory.

Given this set up [330] present the linear Gaussian case, where it is shown that this situation could be solved by the Kalman filter, we will introduce the derivation of the Kalman filer in a later chapter. We refer the reader to [330] for this example.

The next phase of this work is to find a variational equivalent of the analysis step for the linear Gaussian case, again see [330] for details. The part that we do present is that associated with the derivation for the general nonlinear, non-Gaussian case. We provide a warning here that this is a very probability based derivation.

From the linear case in [330] it is shown that assimilating data immediately it becomes available can be cast into a standard signal model/observation model from (16.122) and (16.123), where given these two formulations it is possible to compute the conditional PDFs; $p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_k\right)$ and $p\left(\underline{\mathbf{x}}_{k+1} \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_k\right)$.

In general, according to [330], given the prior PDF $p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_{k-1}\right)$ and the PDF of the observations, given the state, $p\left(\underline{\mathbf{y}}_k \middle| \underline{\mathbf{x}}_k\right)$, Bayes's theorem tells us that the posterior PDF, $p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_k\right)$, is

$$p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_k\right) = \frac{p\left(\underline{\mathbf{y}}_k \middle| \underline{\mathbf{x}}_k\right) p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_{k-1}\right)}{\mathcal{N}}, \tag{16.124}$$

where the normalization $\mathcal{N}$ is given by

$$\mathcal{N} \equiv \int_{\underline{\mathbf{x}}_k \in \mathcal{U}} \left[p\left(\underline{\mathbf{y}}_k \middle| \underline{\mathbf{x}}_k\right) p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \ldots, \underline{\mathbf{y}}_{k-1}\right)\right] d\underline{\mathbf{x}}_k, \tag{16.125}$$

where the domain of integration is $\mathcal{U} = \mathbb{R}^{(N+1)n}$. It is now assumed that the basic process satisfies the Markov property; $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \dots) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$ which implies the same for underlined states; $p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{x}}_{k-1}, \underline{\mathbf{x}}_{k-2}, \dots\right) = p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{x}}_{k-1}\right)$.

Before we progress we have to introduce a definition, and then a named equation to be able follow the derivation in [330].

Definition 16.8. In the theory of stochastic processes in probability theory and statistics, a **nuisance variable** is a random variable that is fundamental to the probabilistic model, but that is of no particular interest in itself, or is no longer of any interest.

Definition 16.9. Suppose that $\{f_i\}$ is an indexed collection of random variables, that is, a stochastic process. Let $p_{i_1,\dots,i_n}(f_1, \dots, f_n)$ be the joint probability density function of the values of the random variables f_1 to f_n. Then the **Chapman–Kolmogorov** equation is

$$p_{i_1,\dots,i_{n-1}}(f_1, \dots, f_{n-1}) = \int_{-\infty}^{\infty} p_{i_1,\dots,i_n}(f_1, \dots, f_n)\, df_n,$$

The expression above could be interrupted as a marginalization over the nuisance variable.

When the stochastic process under consideration is Markovian, the Chapman-Kolmogorov equation is equivalent to an identity on **transition densities**. In the Markov chain setting, is it assumed that $i_1 < \dots < i_n$. Thus, because of the Markov property,

$$p_{i_1,\dots,i_n}(f_1, \dots, f_n) = p_{i_1}(f_1)\, p_{i_2;i_1}(f_2 \mid f_1) \cdots p_{i_n;i_{n-1}}(f_n \mid f_{n-1}),$$

where the conditional probability $p_{i;j}\left(f_i | f_j\right)$ is the transition probability between the times $i > j$, the Chapman–Kolmogorov equation takes the form

$$p_{i_3;i_1}(f_3 \mid f_1) = \int_{-\infty}^{\infty} p_{i_3;i_2}(f_3 | f_2)\, p_{i_2;i_1}(f_2 | f_1)\, df_2. \tag{16.126}$$

Informally, this says that the probability of going from state 1 to state 3 can be found from the probabilities of going from 1 to an *intermediate* or *transition* state 2 and then from 2 to 3, by adding up over all the possible intermediate states 2. It is this property that is used in the next steps of the derivation in [330].

Thus from the theory presented, it is stated in [330] that the prior PDF can be expressed as

$$p\left(\underline{\mathbf{x}}_k \middle| \underline{\mathbf{y}}_0, \dots, \underline{\mathbf{y}}_{k-1}\right) = \int_{\underline{\widetilde{\mathbf{x}}}_{k-1} \in \mathcal{U}} p\left(\underline{\mathbf{x}}_k \mid \underline{\widetilde{\mathbf{x}}}_{k-1}\right) p\left(\underline{\widetilde{\mathbf{x}}}_{k-1} \middle| \underline{\mathbf{y}}_0, \dots, \underline{\mathbf{y}}_{k-1}\right) d\underline{\widetilde{\mathbf{x}}}_{k-1}. \tag{16.127}$$

Now by virtue of the fact that in (16.121) the ith sub-vector of $\underline{\mathbf{h}}_k$ depends only on $\mathbf{x}_{K-N+i-1}$, then the conditional PDF, $p\left(\underline{\mathbf{y}}_k \middle| \underline{\mathbf{x}}_k\right)$ of the observations given the states, factors into

$$p\left(\underline{\mathbf{y}}_k \mid \underline{\mathbf{x}}\right) = p\left(\mathbf{y}_k^{(k)} \mid \mathbf{x}_k\right) \times \cdots \times p\left(\mathbf{y}_{k-N}^{(k)} \mid \mathbf{x}_{k-N}\right), \tag{16.128}$$

where the transition PDF, $p\left(\underline{\mathbf{x}}_k \mid \underline{\widetilde{\mathbf{x}}}_{k-1}\right)$, may be written as

$$p\left(\underline{\mathbf{x}}_k \mid \underline{\widetilde{\mathbf{x}}}_{k-1}\right) \equiv p\left(\mathbf{x}_k \mid \underline{\widetilde{\mathbf{x}}}_{k-1}\right) \delta\left(\mathbf{x}_{k-1} - \widetilde{\mathbf{x}}_{k-1}\right) \cdots \delta\left(\mathbf{x}_{k-N} = \widetilde{\mathbf{x}}_{k-N}\right), \tag{16.129}$$

where we believe that δ is a Dirac delta function.

Thus, combining (16.129) with (16.127) yields

$$
\begin{aligned}
p\left(\underline{\mathbf{x}}_k|\underline{\mathbf{y}}_0,\ldots,\underline{\mathbf{y}}_{k-1}\right) &= \int_{\widetilde{\mathbf{x}}_{k-1}\in\mathcal{U}} p\left(\mathbf{x}_k\,|\widetilde{\mathbf{x}}_{k-1}\right) p\left(\begin{array}{c|c} \widetilde{\mathbf{x}}_{k-N-1} & \\ \widetilde{\mathbf{x}}_{k-N} & \\ \vdots & \underline{\mathbf{y}}_0,\ldots,\underline{\mathbf{y}}_{k-1} \\ \widetilde{\mathbf{x}}_{k-1} & \end{array}\right) d\widetilde{\mathbf{x}}_{k-1}, \\
&= \int_{\widetilde{\mathbf{x}}_{k-N-1}\in\mathbb{R}^n} p\left(\mathbf{x}_k\,|\mathbf{x}_{k-1}\right) p\left(\begin{array}{c|c} \widetilde{\mathbf{x}}_{k-N-1} & \\ \widetilde{\mathbf{x}}_{k-N} & \\ \vdots & \underline{\mathbf{y}}_0,\ldots,\underline{\mathbf{y}}_{k-1} \\ \widetilde{\mathbf{x}}_{k-1} & \end{array}\right) d\widetilde{\mathbf{x}}_{k-N-1}, \\
&= p\left(\mathbf{x}_k\,|\mathbf{x}_{k-1}\right) \int_{\widetilde{\mathbf{x}}_{k-N-1}\in\mathbb{R}^n} p\left(\begin{array}{c|c} \widetilde{\mathbf{x}}_{k-N-1} & \\ \widetilde{\mathbf{x}}_{k-N} & \\ \vdots & \underline{\mathbf{y}}_0,\ldots,\underline{\mathbf{y}}_{k-1} \\ \widetilde{\mathbf{x}}_{k-1} & \end{array}\right) d\widetilde{\mathbf{x}}_{k-N-1}. \qquad (16.130)
\end{aligned}
$$

It is stated in [330] that we may cycle (16.130) and (16.124), (16.128) to obtain the posterior PDF for every k.

This paper goes in to a lot more detail than we can present here, but we wanted to make you aware of this technique that was developed after the first edition, that deals with issues of how to use as much data consistently as possible. While the global forecast may have ran, through this technique better lateral boundary conditions for limited area models could be updated for the higher order model run. We recommend to the reader to read the rest of [330] beyond what we have presented here as there is a vigorous derivation and testing of different configurations of time lags, as well as overlapping windows to optimal use the available data.

16.10 Regularization

In this section we present two different approaches that have been suggested since the last edition of the textbook, to deal with two different problems associated with variational data assimilation. The first approach is derived to treat bias in the data assimilation system through using the Wasserstein metric, [423], that is based upon optimal transport theory [120]; while the second approach uses a L_p-norm regularization to help with sparse solutions, [33], such as those associated with meteorological fronts, [144,145], or sea ice, [15].

16.10.1 Optimal Transport

Up until now we have mainly been presenting the cost functions for 3D and 4D VAR from their Bayesian derivations with PDF definitions, but as we saw in the last section, there is a move to also display the cost function in terms of norms to enable other properties of functional analysis to hold. In [120] the opening sentence of the abstracts states that "Usually data assimilation methods evaluate observation-model misfits using weighted L_2 distances". The next sentence explains that "However, it is not well suited when observed features are present in the model with **position errors**".

In the applications chapter in the first edition we presented a method called displacement vectors which is technique that has been used to start to address displacement errors. This is still in the applications chapter, as there are still concerns about not correctly adjusting the environmental fields to support the structure being moved. We should note, as we do in the applications chapter, that this does not address the fundamental equation of why does the model have the feature in the wrong location? i.e. model error!

Optimal transport was pioneered by Monge (1781), [304], to search for the optimal way of displacing sand tiles into holes of the same volume, minimizing the total cost of displacement, and can be viewed as a transportation problem between two probability measures.

As mentioned above, variational data assimilation can be cast in terms of the L_2 norm, but as shown in [120] in their figure 1 which we have a copy of in Fig. 16.14, when an Euclidean distance is used to measure misfits, it has trouble capturing position errors. In [120] they present two curves, ρ_0 and ρ_1, where the second curve can be seen as the first curve but with a position error. The associated cost function to minimize here is $||\rho - \rho_0||^2 + ||\rho - \rho_1||^2$, where the solution is $\rho_* = \frac{1}{2}(\rho_0 + \rho_1)$, and this solution is plotted in Fig. 16.14, where we can see it does not correct for position error, but instead creates two smaller amplitude curves, that effectively almost conserves the mass between the two curves, but introduces a two more errors at these locations. The green curve is the results from the average of the Wasserstein measure and we can see that it moves the feature to the correct place and conserves the shape.

Given this motivation we now present the theory from [120] to implement this approach into a variational cost function, bearing in mind that in doing so we are no longer consistent with the underlying Bayesian model.

We start with a brief explanation of a **Tikhonov regularization**: Suppose that for a known matrix $\mathbf{A}$ and vector $\mathbf{b}$, we wish to find a vector $\mathbf{x}$ such that $\mathbf{Ax} = \mathbf{b}$. However, if no $\mathbf{x}$ satisfies the equation,

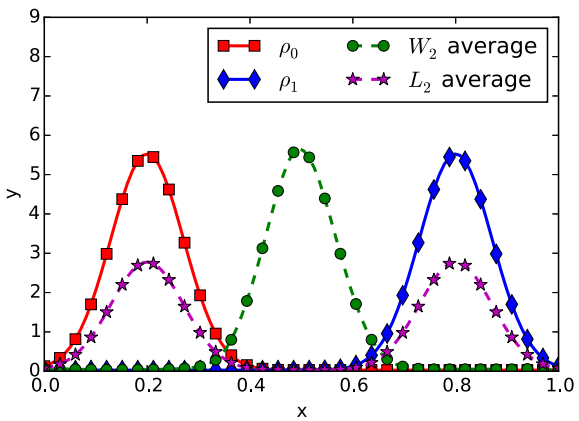


FIGURE 16.14

Copy of figure 1 form Feyeux, N., Vidard, A., and Nodet, M.: Optimal transport for variational data assimilation, Nonlin. Processes Geophys., 25, 55–66, https://doi.org/10.5194/npg-25-55-2018, 2018, https://creativecommons.org/licenses/by/4.0/.

or more than one $\mathbf{x}$ does, then the problem is said to be ill posed. In such cases, ordinary least squares estimation leads to an overdetermined, or more often an underdetermined system of equations.

If we consider a least squares approach, then this methods seeks to minimize the sum of squared residuals, that can be written as $\|A\mathbf{x} - \mathbf{b}\|_2^2$, where $\|\cdot\|_2$ is the Euclidean norm.

In order to give preference to a particular solution with desirable properties, a regularization term can be included in this minimization:

$$\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \|\mathbf{\Gamma}\mathbf{x}\|_2^2, \tag{16.131}$$

for some suitably chosen Tikhonov matrix $\mathbf{\Gamma}$. This regularization improves the conditioning of the problem, thus enabling a direct numerical solution. An explicit solution, denoted by $\hat{\mathbf{x}}$, is given by

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T\mathbf{A} + \mathbf{\Gamma}^T\mathbf{\Gamma}\right)^{-1} \mathbf{A}^T\mathbf{b}. \tag{16.132}$$

The effect of regularization may be varied by the scale of the matrix $\mathbf{\Gamma}$. For $\mathbf{\Gamma} = 0$ this reduces to the unregularized least-squares solution, provided that $\left(\mathbf{A}^T\mathbf{A}\right)^{-1}$ exists.

Therefore, given the explanation above of a Tikhonov regularization, we return to the 4D VAR cost function which is expressed in the following way in [120]:

$$\mathcal{J}(\mathbf{x}_0) = d_R(\mathcal{G}(\mathbf{x}_0), \mathbf{y})^2 + d_B(\mathbf{x}_0, \mathbf{x})^2, \tag{16.133}$$

where d_R and d_B are distances to be specified, and $\mathcal{G} \equiv \mathbf{h} \circ \mathcal{M}$ which is the composition of the nonlinear observation operator with the nonlinear model.

The standard formulation for 4D VAR is to replace the two distances with the L_2 norm. However, in [120] they propose to replace this with the Wasserstein distance instead, which we present now. It is assumed that the observations can be represented as positive fields that are referred to as mass functions, that are non-negative functions of space, that have the following definition from [120]:

Definition 16.10. Let Ω be a closed, convex, bounded set of $\mathbb{R}^d$ and let the set of mass functions $\mathcal{P}(\Omega)$ be the set of non-negative functions of total mass 1:

$$\mathcal{P}(\Omega) := \left\{\rho \geq 0 : \int_\Omega \rho(x)\, dx = 1\right\}. \tag{16.134}$$

In the mathematical framework of optimal transport, mass functions are continuous and are called "probability densities". In the data assimilation framework the concept of probability densities is mostly used to represent errors. Here, the positive functions we consider serve as observations or state vectors, so [120] chose to call them mass functions to avoid any possible confusion with state or observation error probability distributions.

We now present the Wasserstein distance from [120]. Thus, given the set of all transportations between two mass functions, the optimal transport is the one that minimizes the kinetic energy. A transportation between two mass functions, ρ_0 and ρ_1, is given by a time path $\rho(t, x)$ such that $\rho(t = 0) = \rho_0$ and $\rho(t = 1) = \rho_1$ and given by a velocity field $\mathbf{v}(t, x)$, such that the continuity equation,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{x}) = 0, \tag{16.135}$$

holds. Thus this path $\rho(t)$ can be seen as interpolating ρ_0 and ρ_1. For $\rho(y)$ to stay in $\mathcal{P}(\Omega)$, a sufficient condition is that the velocity field $\mathbf{v}(t,x)$ should be tangent to the domain boundary, meaning that $\rho(t,x)\mathbf{v}(t,x)\cdot\mathbf{n}(x)=0$ for almost all $(t,x)\in[0,1]\times\partial\Omega$. Note that time here is fictitious.

Therefore, the Wasserstein distance $\mathcal{W}$ is the minimum in terms of kinetic energy amongst all the transportations between ρ_0 and ρ_1, given by

$$\mathcal{W}(\rho_0,\rho_1)=\sqrt{\min_{(\rho,\mathbf{v})\in C(\rho_0,\rho_1)}\iint_{[0,1]\times\Omega}\rho(t,x)|\mathbf{v}(t,x)|^2\,dtdx} \tag{16.136}$$

where in [120] it is stated that $C(\rho_0,\rho_1)$ represents the set of continuous transportations between ρ_0 and ρ_1 described by a velocity field $\mathbf{v}$ tangent to the boundary of the domain:

$$C(\rho_0,\rho_1)=\left\{(\rho,\mathbf{v})\,s.t.\ \begin{array}{l}\dfrac{\partial\rho}{\partial t}+\nabla\cdot(\rho\mathbf{v})=0,\\ \rho(t=0)=\rho_0,\quad \rho(t=1)=\rho_1,\\ \rho\mathbf{v}\cdot\mathbf{n}=0\quad\text{on}\quad\partial\Omega.\end{array}\right. \tag{16.137}$$

It is stated in [120] that the optimal velocity field $\mathbf{v}$ is of the form $\mathbf{v}=\nabla\Phi(t,x)$ where Φ satisfies the Hamilton-Jacobi equation

$$\frac{\partial\Phi}{\partial t}+\frac{|\nabla\Phi|}{2}=0. \tag{16.138}$$

Thus the optimal ρ is the continuity equation using this velocity field. We state now that the rest of the derivation from [120] relies on a lot of properties from functional analysis, that we do not have the space to explain, but there are several really good textbooks out there to explain these terms. The main aim of this section is to introduce new techniques that help improve the performance of the variational systems, and provide the proofs where we can.

The function Φ defined by $\Psi=-\Phi(t=0,x)$ is said to be the *Kantorovich potential*, of the transport between ρ_0 and ρ_1, that allows the computation of the Wasserstein distance, to be

$$\mathcal{W}(\rho_0,\rho_1)^2\equiv\int_\Omega\rho_0(x)|\nabla\Psi(x)|^2\,dx. \tag{16.139}$$

The next tool to define is the Wasserstein inner product, where in [120] it is stated that scalar product defines the angle and norm of vectors tangent to $\mathcal{P}(\Omega)$ at a point ρ_0, where a tangent vector in ρ_0 is the derivative of a curve of $\rho(t)$ passing through ρ_0. As a curve $\rho(t)$ can be described by a continuity equation, the space of tangent vectors (tangent space), is defined by

$$T_{\rho_0}\mathcal{P}=\left\{\begin{array}{l}\eta\in\mathcal{L}^2(\Omega),\\ \eta=-\nabla\cdot(\rho_0\nabla\Phi),\\ \Phi\quad\text{s.t.}\quad\rho_0\dfrac{\partial\Phi}{\partial\mathbf{n}}=0\quad\text{on}\quad\partial\Omega.\end{array}\right. \tag{16.140}$$

The Wasserstein inner product $\langle\cdot,\cdot\rangle_{\mathcal{W}}$ is defined for $\eta=-\nabla\cdot(\rho_0\nabla\Phi)$, $\eta'=-\nabla\cdot\left(\rho_0\nabla\Phi'\right)\in T_{\rho_0}\mathcal{P}$ by

$$\langle\eta,\eta'\rangle_{\mathcal{W}}\equiv\int_\Omega\rho_0\nabla\Phi\cdot\nabla\Phi' dx. \tag{16.141}$$

Thus the norm associated with the tangent vector $\eta = -\nabla \cdot (\rho_0 \nabla \Phi) \in T_{\rho_0}\mathcal{P}$ is

$$\|\eta\|_{\mathcal{W}}^2 = \int_\Omega \rho_0 |\nabla \Phi|^2 \, dx, \tag{16.142}$$

and in [120] they state that this represents the kinetic energy of the small displacements.

We note here that there is a lot of discussions, as well as a theorem, presented in [120] to do with the gradient of the Wasserstein cost function which is again too detailed for us to present here; so please see [120] for all these details; what we do present here is the alternative cost function to the standard $\mathcal{L}_2$ cost function which

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2} \sum_{i=1} N_o \mathcal{W}(\mathcal{G}_i(\mathbf{x}_0), \mathbf{y}_i)^2 + \frac{\omega_b}{2} \mathcal{W}\left(\mathbf{x}_0, \mathbf{x}_{b,0}\right)^2, \tag{16.143}$$

where ω_b is a scalar weight associated with the background term. In [120] this approach is tested with a linear evolution model and a nonlinear shallow water equations model and the reader is referred to this paper to see this results.

The method presented here is to address displacement errors, but it does not reduce the bias if this is present as well. To address this point we now consider the work from [423].

To address the point that just made, in [423] they indicate that a convex relaxation of the optimal mass transport from Monge through probabilistic consideration of transport, in which mass at any source point x_i can be *split* and transported across several target points z_i is possible. In [423] there is an illustration of this approach and we have a copy of this figure in Fig. 16.15, but also shows the difference between this approach and that of Monge. This approach is also seen as an alternative to the cumulative density function (CDF) matching methods, where if we let $\mathbf{p}_x = \left(p_{x_1}, \ldots, p_{x_k}\right)^T \in \mathbb{R}^k$ and $\mathbf{p}_z = \left(p_{z_1}, \ldots, p_{z_l}\right)^T \in \mathbb{R}^l$ represent the probability vectors associated with a source and a target histogram supported on vectors $x_1.x_2, \ldots, x_k$ and $z_1, z_2, \ldots, z_l$ respectively. In the Monge formulation the problem involves seeking a surjective optimal transport mapping $T : \{x_1, x_2, \ldots, x_k\} \mapsto \{z_1, z_2, \ldots, z_l\}$ that moves probability mass from each discrete point, x_i, on the source histogram to a single point z_j on the target probability histogram such that the total cost of transportation is minimized:

$$\begin{aligned} \min_{T(\cdot)} \quad & \sum_i c(x_i, T(x_i)) \\ \text{subject to} \quad & p_{z_i} = \sum_{i:T(x_i)=z_i} p_{x_i}, \end{aligned} \tag{16.144}$$

where $c(x_i, T(x_i))$ is the transportation cost between points x_i and $T(x_i)$ and the constraint warrants the mass conservation principle.

Returning to the theory proposed in [423], we now assume that $\mathbf{U} \in \mathbb{R}^{k \times l}$ represents a *transportation plan matrix* where u_{ij} describe the probability mass being transferred from point x_i to point z_i, and $\mathbf{C} \in \mathbb{R}^{k \times l}$ represents the transportation or ground cost matrix, where the elements $c_{i,j} = \left|x_i - z_j\right|^p$, is the cost of transporting probability masses from x_i to z_j for $i = 1, 2, \ldots . k$ and $j = 1, 2, \ldots, l$, and p is a positive exponent, and so this is now a Wasserstein distance, $\mathcal{W}_p$ that seeks to minimize the total amount of work done in transporting probability masses $\mathbf{p}_x$ to $\mathbf{p}_z$ as:

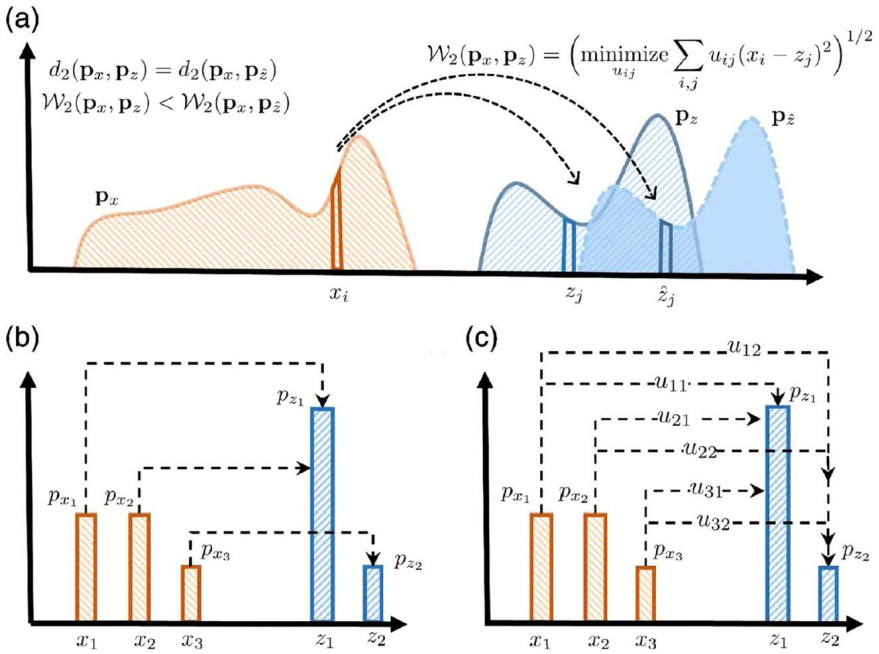


FIGURE 16.15

Copy of figure 1 from [423].

$$\begin{aligned}\mathcal{W}_p(\mathbf{p}_x.\mathbf{p}_z) &\equiv \min_{u_{ij}} \left(\sum_{ij} c_{ij}u_{ij}\right)^{\frac{1}{p}}, \\ &= \min_{\mathbf{U}} \left(\text{tr}\mathbf{C}^T\mathbf{U}\right)^{\frac{1}{p}}, \\ &\text{subject to } u_{ij} \geq 0, \quad \mathbf{U}\mathbf{1}_l = \mathbf{p}_x. \quad \mathbf{u}^T\mathbf{1}_k = \mathbf{p}_z,\end{aligned} \tag{16.145}$$

where the first constraint ensures that the transport probability masses are non-negative, whilst the second and third constraint enforce the conservation of mass. Thus the Wasserstein metric between two probability histograms is the minimum cost required to match them through a transportation plan matrix. It is stated in [423] that the Wasserstein metric penalizes the misfit between the shape of the histograms and increase monotonically with the shift between central positions of the histograms, enabling it to penalize bias naturally. It is then claimed, without proof, that for $p = 2$ we have $\mathcal{W}_2^2(\mathbf{p}_x, \mathbf{p}_z) = \mathcal{W}_2^2(\widetilde{\mathbf{p}}_x, \widetilde{\mathbf{p}}_z) + \|\boldsymbol{\mu}_x - \boldsymbol{\mu}_z\|$, where $\widetilde{\mathbf{p}}_x$ and $\widetilde{\mathbf{p}}_z$ are the centered zero-mean probability masses and $\boldsymbol{\mu}_x$ and $\boldsymbol{\mu}_z$ are the mean values.

Thus the derivation to the regularization of variational data assimilation through the Wasserstein metric (WM-VDA) is obtained as follow: The WM-VDA cost function is defined as

$$\mathcal{J}_{WM-VDA}(\mathbf{x}, \mathbf{p}_x) = \|\mathbf{x} - \mathbf{x}_b\|^2_{B^{-1}} + \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2_{\mathbf{R}^{-1}} + \lambda\mathcal{J}_{\mathcal{W}_2}(\mathbf{p}_x, \mathbf{p}_z), \tag{16.146}$$

where $\mathcal{J}_{\mathcal{W}_2}(\mathbf{p}_x, \mathbf{p}_z)$ represents the transportation cost associated with the square of the 2-Wasserstein distance between the two probability histograms and λ is a non-negative regularization parameter, that balances a trade-off between the Euclidean and the Wasserstein cost.

Unfortunately, the next step in this derivation restricts this approach to Gaussian distribution in the statement that the analysis state at the initial time is an *expected value*. This cost function without the regularization term is Gaussian based, so its transferability to other distributions can not be taken as a straightforward crossover. However, following the format in [423], the initial state can be represented as $\mathbf{x} = \mathbf{X}\mathbf{p}_x$ which results in the following cost function:

$$\mathcal{J}_{WM-VDA}(\mathbf{p}_x) = \|\mathbf{X}\mathbf{p}_x - \mathbf{x}_b\|^2_{B^{-1}} + \|\mathbf{y} - \mathbf{H}\mathbf{X}\mathbf{p}_x\|^2_{\mathbf{R}^{-1}} + \lambda\mathcal{J}_{\mathcal{W}_2}(\mathbf{p}_x, \mathbf{p}_z). \tag{16.147}$$

Applying the mass constraint from earlier, we have $\mathbf{p}_x = \mathbf{U}\mathbf{1}_{k^m}$, and setting $\mathcal{J}(\mathbf{p}_x, \mathbf{p}_z) = \text{tr}\left(\mathbf{C}^T\mathbf{U}\right)$, then the cost function above can be expressed in terms of the transportation matrix, $\mathbf{U}$, as

$$\mathcal{J}_{WM-VDA}(\mathbf{U}) = \|\mathbf{X}\mathbf{U}\mathbf{1}_{k^m} - \mathbf{x}_b\|^2_{B^{-1}} + \|\mathbf{y} - \mathbf{H}\mathbf{X}\mathbf{U}\mathbf{1}_{k^m}\|^2_{\mathbf{R}^{-1}} + \lambda\text{tr}\left(\mathbf{C}^T\mathbf{U}\right). \tag{16.148}$$

In [423] they present a discussion about solving (16.148) through quadratic programming, which is again beyond the scope of this textbook, and the reader is referred to this paper for the details, as well promising results from this approach with the Lorenz 1963 model.

As we have seen over this subsection, there is a lot of research to improve the performance of the variational systems, but we add the caveat here that when moving into functional analysis space, we lose the probability structure, even those there are links back to the Gaussian distribution, there isn't to say even the lognormal distribution and as such it comes back to the question about the displacement vector approach. If the Gaussian Bayesian approach is not the most suitable model for the errors, then consider other distributions in the Bayes theorem model, which the author has spent the last 18 years, at time of writing, working on.

This aside the next regularization approach we present deals with resolving sharper features rather than displacement errors or biases.

16.10.2 L_p-Norm Regularization

In the late 1990s and early 2000s when undertaking our Ph.D. at the University of Reading under the supervision of Professor Nancy Nichols, and Dr. Ian Roulstone, now Professor, these regularization terms were referred to as the J_c terms, where these were extra constraints added to the cost function to ensure balance was enforced on the initial conditions, we explain this a bit more in the next chapter, but this is to put this into a historical context. Thus the variational formulation has this flexibility to add more constraint. We should note that this is beyond the Bayes theorem model, but when forming a calculus of variation problem with functionals, as we saw earlier, we can add as many constraints on the solution as are needed to find the optimal solution.

The constraint that is introduced in [33] is to help with resolving sharp features, discontinuities, or jumps, in the flow that produce spares solutions. The first point that is made in [33] is that the Tikhonov regularization (the background term) usually results in smooth solutions.

The derivation of the L_p regularization presented in [33] is motivated and derived as follows: Because sparsity is a strong property of a system, then incorporating a priori information on sparsity – if

available – in the 4D-Var minimization problem could be beneficial. According to [33] the most used a priori information for promoting sparsity is the generalized Gaussian distribution:

$$p_p(\mathbf{x}) \sim \exp\left\{-\sum_{i=1}^{N} |(\Phi x_i)|^p\right\} \tag{16.149}$$

where Φ in [33] is referred to as the basis in which the vector $\mathbf{x}$ is sparse and $p \in [0, 2]$. The choice of p is critical in determining the character of the model: larger values of p discourage abrupt discontinuities while smaller values of p foster them [33]. Specifically:

- If $p \in [0\ 0, 1)$ the L_p-norm is not a convex function, thus the Bayesian model we have for variational data assimilation has several local minima.
- If $p = 1$, then the L_1-norm minimization problem is obtained and introducing an L_1-norm penalty term promotes sparsity. From [33] there appears to ve quite a few problems considering this approach, but the reader is referred to the references in [33] for more details about this.
- The non-smooth a priori information $p_1(\mathbf{x})$ can be replaced with a smoothed version $p_p(\mathbf{x})$ with $p \in (1, 2)$. According to [33]; in this case, optimization methods dedicated to convex functionals can be applied. If p is close to 1, the L_p-norm promotes sparsity like the L_1-norm. Increasing p gradually from 1 to 2 leads simultaneously to a decrease of the penalty on 'small' coefficients, (those with $\left|\Phi x_j\right| < 1$) and an increase of the penalty on 'large' coefficients. Smoother and less sparse solutions are thus obtained when increasing p.
- If $p = 2$, then the classical L2-norm is obtained and the function (8) is Gaussian. This regularization term tends to impose extra smoothness on the solution.
  Therefore, [33] reformulate the 4D VAR cost function with the extra constraint that the a priori information is sparse as

$$\mathcal{J}_p(\mathbf{x}) \equiv \min_{\mathbf{x}} \left\{\frac{1}{2}\|\mathbf{x} - \mathbf{x}_{0,b}\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\|\underline{\mathbf{y}} - \underline{\mathbf{h}}(\underline{\mathbf{x}})\|^2_{\mathbf{R}^{-1}} + \frac{\lambda}{p}\|\Phi\mathbf{x}\|_p^p\right\}. \tag{16.150}$$

  In [33] they go on to present a statistical interpretation of their approach, along with a functional analysis explanation of their minimization approach to the cost function above, see that paper for more details.

16.11 4D VAR as an Optimal Control Problem

We now present a summary of the work from [233] associated with linking 4D VAR to an optimal control problem. We start with the discrete time nonlinear model dynamics $\mathcal{M}(\boldsymbol{x}_k, \boldsymbol{\eta}_k)$ such that

$$\overline{\boldsymbol{x}}_{k+1} = \overline{\mathcal{M}}(\overline{\boldsymbol{x}}_k, \boldsymbol{\eta}_k), \tag{16.151}$$

where $\overline{\mathcal{M}} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$, is the state of the time-invariant dynamics, $\boldsymbol{x}_k \in \mathbb{R}^n$ is the **state** vector of the time-invariant dynamics, and $\boldsymbol{\eta}_k \in \mathbb{R}^n$ is a given intrinsic physical forcing that is part of the model.

As we mentioned earlier, one of the advantages of the variational approach is that we can add as many constraints to help find the best solution for the initial conditions, and in this section we have presented three different approaches to deal with displacement errors, bias, and spurious features in the

numerical solution, which from what we can see in [33] appears to be the goal here. There are a lot of details missing from this, and the other papers referred to in this section, and we have tried to fill in as many details as possible.

This brings us to the end of the direct theory of variational base data assimilation, but before we move onto deal with different aspects of implementing the variational systems, we prove that 4D VAR can be viewed as an optimal control problem.

The derivation of 4D VAR as an optimal control problem is based on the Pontryagin minimum principle and requires an external force, or control, to the given dynamics in (16.151). Therefore we can rewrite (16.151) as

$$\boldsymbol{x}_{k+1} = \mathcal{M}\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k, \boldsymbol{u}_k\right) = \overline{\mathcal{M}}\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k\right) + \mathbf{B}\boldsymbol{u}_k. \tag{16.152}$$

Note that the matrix $\mathbf{B}$ here is as it was defined in the control theory chapter, and is not the background error covariance matrix. We know that the control $\boldsymbol{u}_k$ is usually of a lower dimension than that of the model. We now introduce the observation vector, $\boldsymbol{y}_k$, at time k, defined as

$$\boldsymbol{y}_k = \boldsymbol{h}\left(\boldsymbol{x}_k\right) + \boldsymbol{\varepsilon}_k, \tag{16.153}$$

where $\boldsymbol{y}_k \in \mathbb{R}^m$ for some positive integer m, and $\boldsymbol{\varepsilon}_k$ is the observational error which is assumed to be Gaussian distributed such that $\boldsymbol{\varepsilon}_k \sim N\left(\mathbf{0}, \mathbf{R}\right)$.

We now define a performance measure as

$$J = \sum_{k=0}^{N-1} V_k\left(\boldsymbol{x}_k, \boldsymbol{y}_k, \boldsymbol{u}_k\right), \tag{16.154}$$

where N is the number of observations and the cost function V_k is the sum of two terms given by

$$V_k\left(\boldsymbol{x}_k, \boldsymbol{y}_k, \boldsymbol{u}_k\right) = V_k^0\left(\boldsymbol{x}_k, \boldsymbol{y}_k\right) + V_k^c\left(\boldsymbol{u}_k\right), \tag{16.155}$$

where

$$V_k^0\left(\boldsymbol{x}_k, \boldsymbol{y}_k\right) = \frac{1}{2}\left\langle\left(\boldsymbol{y}_k - \boldsymbol{h}\left(\boldsymbol{x}_k\right)\right), \mathbf{R}^{-1}\left(\boldsymbol{y}_k - \boldsymbol{h}\left(\boldsymbol{x}_k\right)\right)\right\rangle, \tag{16.156a}$$

$$V_k^c\left(\boldsymbol{u}_k\right) = \frac{1}{2}\left\langle \boldsymbol{u}_k, \mathbf{C}\boldsymbol{u}_k\right\rangle, \tag{16.156b}$$

and $\langle\cdot,\cdot\rangle$ is the standard inner product and $\mathbf{C} \in \mathbb{R}^{p\times p}$ is a given symmetric positive definite matrix. It is stated in [233] that V_k^0 denotes the energy in the normalized forecast errors

$$\boldsymbol{e}_k = \boldsymbol{y}_k - \boldsymbol{h}\left(\boldsymbol{x}_k\right), \tag{16.157}$$

and V_k^c accounts for the energy in the control input.

We now defined the Lagrangian, $\mathcal{L}$, which is obtained by augmenting the dynamical constraint in (16.151) with the cost function in (16.154) as

$$\mathcal{L} = \sum_{k=0}^{N-1}\left(V_k + \left\langle\boldsymbol{\lambda}_{k+1}, \left(\mathcal{M}\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k, \boldsymbol{u}_k\right) - \boldsymbol{x}_{k+1}\right)\right\rangle\right), \tag{16.158}$$

where $\boldsymbol{\lambda}_k$ for $k = 1, 2, \ldots, N$ denotes the set of N undetermined Lagrangian multipliers. Now we define the **Hamiltonian**, H_k, function as

$$H_k = H_k\left(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\eta}_k, \boldsymbol{\lambda}_k\right) = V_k + \langle \boldsymbol{\lambda}_{k+1}, \mathcal{M}\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k, \boldsymbol{u}_k\right)\rangle. \tag{16.159}$$

Given the definition of the Hamiltonian for this problem, we can rewrite the Lagrangian functional as

$$\mathcal{L} = \sum_{k=0}^{N-1}\left(H_k - \langle \boldsymbol{\lambda}_{k+1}, \boldsymbol{x}_{k+1}\rangle\right). \tag{16.160}$$

By taking the term associated with $k = N$ from the summation in (16.160) and H_0 (we shall see this trick again in Chapter 18), we have

$$\mathcal{L} = H_0 - \langle \boldsymbol{\lambda}_N, \boldsymbol{x}_N\rangle + \sum_{k=1}^{N-1}\left(H_k - \langle \boldsymbol{\lambda}_{k+1}, \boldsymbol{x}_{k+1}\rangle\right). \tag{16.161}$$

It is stated in [233] that there are now variations in $\boldsymbol{\eta}_k$ due to this parameter being specified. Therefore we shall only consider variations $\delta\boldsymbol{x}_k$ and $\delta\boldsymbol{u}_k$ with respect to $\boldsymbol{x}_k$ and $\boldsymbol{u}_k$, respectively, as well as variations $\delta\boldsymbol{\lambda}_k$ in $\boldsymbol{\lambda}_k$ for $0 \leq k \leq N$. Therefore, through taking the first variation of H_k we obtain

$$\begin{aligned}\delta\mathcal{L} = \langle \nabla_{\boldsymbol{x}_0} H_0, \delta\boldsymbol{x}_0\rangle + \langle \nabla_{\boldsymbol{u}_0} H_0, \delta\boldsymbol{u}_0\rangle - \langle \boldsymbol{\lambda}_N, \delta\boldsymbol{x}_N\rangle + \sum_{k=1}^{N}\langle \nabla_{\boldsymbol{\lambda}} H_{k-1} - \boldsymbol{x}_k, \delta\boldsymbol{\lambda}_k\rangle \\ + \sum_{k=1}^{N-1}\langle \nabla_{\boldsymbol{x}} H_k - \boldsymbol{\lambda}_k, \delta\boldsymbol{x}_k\rangle + \sum_{k=1}^{N-1}\langle \nabla_{\boldsymbol{u}} H_k, \delta\boldsymbol{u}_k\rangle,\end{aligned} \tag{16.162}$$

where $\nabla_{\boldsymbol{x}} H_k \in \mathbb{R}^n$, $\nabla_{\boldsymbol{u}} H_k \in \mathbb{R}^p$, and $\nabla_{\boldsymbol{\lambda}} H_k \in \mathbb{R}^n$ are the Jacobians of the kth Hamiltonian H_k, with respect to $\boldsymbol{x}_k$, $\boldsymbol{u}_k$, and $\boldsymbol{\lambda}_{k+1}$, respectively.

Now we recall the theory from the optimal control chapter that we require $\delta\mathcal{L}$ to be zero for all admissible variations; therefore, we collect the factors of the different variations in (16.162) and consider each one in turn for the condition just mentioned. Therefore, the first condition is

$$\begin{aligned}\boldsymbol{x}_k = \nabla_{\boldsymbol{\lambda}} H_{k-1} = \frac{\partial H_{k-1}}{\partial \boldsymbol{\lambda}_k} = \mathcal{M}\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k, \boldsymbol{u}_k\right), \\ = \overline{\mathcal{M}}\left(\boldsymbol{x}_{k-1}, \boldsymbol{\eta}_{k-1}\right) + \mathbf{B}\boldsymbol{u}_{k-1}\end{aligned} \tag{16.163}$$

which is referred to as condition 1 in [233], which is the **model dynamics**. This is what we saw in Chapter 7.

We now consider the summation with respect to the variations of the model variables, such that

$$\boldsymbol{\lambda}_k = \nabla_{\boldsymbol{x}} H_k = \frac{\partial H_k}{\partial \boldsymbol{x}_k}. \tag{16.164}$$

Therefore, calculating the gradient of the Hamiltonian, as defined in (16.159), with respect to the model variables, results in

$$\boldsymbol{\lambda}_k \mathbf{M}^T \boldsymbol{\lambda}_{k+1} + \nabla_{\boldsymbol{x}} V_k, \tag{16.165}$$

where $\nabla_{\boldsymbol{x}} V_k$ is given by

$$\nabla_x V_k = \nabla_{\boldsymbol{x}} V_k^0 = \mathbf{H}^T \mathbf{R}^{-1} \left(\boldsymbol{h}\left(\boldsymbol{x}_k\right) - \boldsymbol{y}\right), \tag{16.166}$$

and $\mathbf{H}$ and $\mathbf{M}$ are the tangent linear models for the observation and model operators as defined earlier, where the transpose of these matrices are their adjoints. Thus by collecting the terms in (16.165) and (16.166) we obtain the **adjoint dynamics**, which are given by

$$\boldsymbol{\lambda}_k = \mathbf{M}^T \boldsymbol{\lambda}_{k+1} + \mathbf{H}^T \mathbf{R}^{-1} \left(\boldsymbol{h}\left(\boldsymbol{x}_k\right) - \boldsymbol{y}_k\right). \tag{16.167}$$

We do not need to consider the variations with respect to the control for 4D VAR, because 4D VAR comes about through considering the unforced case of the description above. However, we do need to deal with the transversality conditions. We know that we have initial conditions to start the model with and as such the first inner product in (16.162) can only be zero if $\delta \boldsymbol{x}_0$ is zero. Therefore, ignoring the second inner product, we have the third inner product in (16.162) to consider. Because we do **not** know the model at the end of the window we cannot set $\delta \boldsymbol{x}_N$ equal to zero, which means $\boldsymbol{\lambda}_N = \mathbf{0}$.

Therefore, the remaining term is $\langle \nabla_{\boldsymbol{x}_0} H_0, \delta \boldsymbol{x}_0 \rangle$ can be shown to be

$$\begin{aligned} &\delta \mathcal{L} = \langle \nabla_{\boldsymbol{x}_0} H_0, \delta \boldsymbol{x}_0 \rangle \\ \Rightarrow &\frac{\partial \mathcal{L}}{\boldsymbol{x}_0} = \nabla_{\boldsymbol{x}_0} H_0 = \mathbf{M}_1 \boldsymbol{\lambda}_1 + \mathbf{H}\mathbf{R}^{-1} \left(\boldsymbol{h}\left(\boldsymbol{x}_0\right) - \boldsymbol{y}\right)k). \end{aligned} \tag{16.168}$$

We have to note from (16.167) that $\boldsymbol{\lambda}_1$ is a function of $\boldsymbol{\lambda}_2$ and so on, and therefore we have that Lagrangian is minimized through

$$\begin{aligned} &\boldsymbol{\lambda}_N = \mathbf{0}, \\ &\boldsymbol{\lambda}_k = \mathbf{M}_k^T \boldsymbol{\lambda}_{k+1} + \mathbf{H}_k^T \mathbf{R}^{-1} \left(\boldsymbol{h}\left(\boldsymbol{x}_k\right) - \boldsymbol{y}_k\right), \end{aligned} \tag{16.169}$$

which is the same as the gradient of the full field 4D VAR cost function and hence 4D VAR is equivalent to an optimal control problem through the Pontryagin minimum principle.

Given the progress in computer architecture we presented an approach designed to parallelize in time the 4D VAR cost functions, referred to as the saddle point approach. This is part of the new work added to this chapter from the first edition.

16.12 Summary

In this chapter we have introduced the theory of variational data assimilation. We started with a summary of the pioneering work of Yoshikazu Sasaki from his (1970) paper that introduced the terminology of **weak** and **strong** constraints to represent the model error and the perfect model situations, respectively, on what would eventually become 4D VAR.

We introduced the full field 3D VAR formulation from the Bayesian approach from Lorenc's 1986 paper, where we showed that it was possible to find three different expressions for the linear solution to the 3D VAR cost-function.

The derivation of 4D VAR presented was as a result of the work of a series of papers but started with Lewis and Derber in 1985; where they used the adjoint to find a trajectory that fitted a set of observations in a least squares sense.

Next we moved on to the theory that made 4D VAR operationally viable, which was the introduction of the **incremental formulation** by Courier et al. in (1994). Through assuming that the background state is a quite good approximation to the true state, and that we are only seeking a small change to the background to be able to obtain the true state, then we only require a linear model for the evolution of this increment. We showed that the linearization of the cost functions is achieved through the tangent linear approximation and that the adjoint is simply the transpose of the tangent linear model.

We derived the incremental 3D and 4D VAR cost functions and mentioned that the model error and the tangent linear assumption holding were restrictions on how long the assimilation window could be. We introduced the concepts of the inner and outer loops in the minimization of the cost functions. We also briefly introduced two different approaches to implement 3D-FGAT as a way to incorporate some temporal information into the 3D VAR cost function.

Given the restrictions of the model error, we introduced some of the suggested approach from Trèmolet's 2006 and 2007 papers, where we briefly mentioned the modeling required for a good estimate of the model error covariance matrix $\mathbf{Q}$. We have also introduced the more recent work at ECMWF to address time correlated model errors, through a time lagged approach. When we have a good choice for the $\mathbf{Q}$ matrix, it is possible to extend the window length in a weak constraint formulation compared to the strong constraint.

We also introduced the work done to identify the correlated observational errors, and the process of starting to fill in the $\mathbf{R}$ matrix, rather than compensating with inflated variances on the diagonal, which was the case before centers started to approximate the correlations. We also made reference to work that is being done to quantify the other component of the observational error which is the representative error.

In this edition of the textbook we have introduced the new topics of FSOI, saddle-point 4D VAR to allow for the parallelization in time of 4D VAR, overlapping window assimilation, as well as different regularization techniques to aid the variational data assimilation systems to have better solutions.

We finished this chapter with the mathematical proof that the 4D VAR cost function could be derived from the Pontryagin minimum principle from control theory.

Variational-based data assimilation schemes are still widely used as operational system for many different geoscience disciplines, but we should note the formulations that we have presented in this chapter are referred to as **static** because of the model used for the background error covariance matrix, $\mathbf{B}_0$. We now move on to summarize the different subcomponents that are required to implement VAR in different geophysical disciplines.

This page intentionally left blank

CHAPTER

Subcomponents of Variational Data Assimilation

17

Contents

In the last chapter we introduced variational-based data assimilation methods that have been used for operational numerical weather and ocean prediction since the early 1990s. However, as we saw there are many different components that have to be addressed to be able to implement the variational methods. In this chapter, we shall introduce the concept of balance and initialization, where we are trying to constrain the analysis state exciting spurious waves, or unphysical solutions, in the numerical model forecast from that state.

One of the drawbacks of the variational methods is the appearance of the inversion of the large background error covariance matrix. Some of the operational numerical weather prediction centers

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00021-8

now run their forecast models at resolutions that make the background error covariance matrix be of the order of 10^9, this would make $\mathbf{B}_0$ of dimensions $10^9 \times 10^9$, which is almost impossible to store, let alone invert. Another problem with the **B** matrix is that it is quite full, and as such we cannot apply sparse matrix inversions, or system of simultaneous equation solvers. Therefore, we shall introduce techniques later that are used to reduce the dimension of this matrix, but also to decorrelate variables to make the matrix more sparse.

The process of transforming the state variables into less correlated variables is referred to as the **control variable transform**. This is to indicate that we are not minimizing the cost function with respect to the model variable, but with respect to the **control variables**. There has been much research to try to determine what are the best set of variables to use for various different geophysical problems. For the spectral-based methods the obvious choices are streamfunction and velocity potential, as the spherical harmonics are based on the Laplacian operator, which is how these two variables are related to the horizontal wind components. Other motivations for the choice of control variables is to try to introduce some flow dependency into these variables. The formulation that we introduced in the last chapter was the **static** formulation of VAR. Given a static background error covariance matrix, the application of the variational methods would have the same background error covariance matrix for all different dynamical situations, which is not optimal. To overcome the suboptimality of the static approach, before the ensemble methods became widespread, a possible approach was to use more dynamical control variables. We shall present sets of approaches that were considered; the first was to use potential vorticity instead of the relative vorticity as the control variable, while another approach was to expand the wind field through an asymptotic expansion of the horizontal wind field [128,287, 288].

After the control variables have been selected to ensure that the specific balance is achieved, we then have to determine how to model the covariances between these variables, and so we shall introduce different covariance models that have been used at different times to model the covariances, but also techniques to determine **decorrelation lengths**.

After determining the covariance model that is to be used, we introduce different preconditioning techniques that are applied to help accelerate the minimization of the iterative solver that is used; we shall finish this chapter with a summary of different minimization algorithm and metrics that are used to assess the performance of different data assimilation schemes.

17.1 Balance

In Chapter 10 we looked at different schemes for the advection of buoyancy and potential vorticity in the Eady model. What we did not go into to too much detail about the model was that is based upon the three balances; geostrophic, hydrostatic, and the thermal wind. We now look at a bit more details about how some of these balances can be used to help the variational data assimilation schemes.

17.1.1 Linear and Nonlinear Balances

Balance plays a vital part in keeping a data assimilation scheme stable; the aim of either a linear, or nonlinear, normal mode initialization is to damp, or remove, parts of the initial conditions that will excite spurious motions not present in the geophysical system we are modeling and trying to predict. Before

normal mode initialization, there are a series of balance equations that were derived from scale analysis of the primitive equations, or the shallow water equations, that enable *balances* between different variables.

We shall not derive these balances here, but refer the readers to Roger Daley's 1991 book [85], where there is a detailed derivation of the following balances. However we shall briefly present the equations and mention the type of balance associated with them.

The first balance relationship that we consider we have already met in Chapter 10, and that is **geostrophic balance**. Geostrophic balance, or an initialization constraint as it is referred to in [85], is a first-order approximation to the wind fields and in Cartesian and spherical coordinates is defined by

$$v_g = \frac{1}{f_0}\frac{\partial \Phi}{\partial x}, \qquad u_g = -\frac{1}{f_0}\frac{\partial \Phi}{\partial y}, \tag{17.1a}$$

$$v_g = \frac{1}{f_0 a \cos\theta}\frac{\partial \Phi}{\partial \lambda}, \quad u_g = -\frac{1}{f_0 a}\frac{\cos\theta \Phi}{\partial \theta}, \tag{17.1b}$$

where f_0 is a constant value for the Coriolis parameter, usually set to a value in the mid-latitudes, Φ is the geopotential, and is related to the streamfunction, ψ, through $\Phi = f_0\psi$.

Geostrophic balance represents the rotational component of the wind fields. However, given this balance expression, it is possible to constrain the geopotential so that it satisfies what is referred to as the **linear balance equation**. As mentioned, we assume a constant value for the Coriolis parameter; this is in part to avoid having to differentiate this parameter with respect to the y direction, θ in spherical coordinates. A drawback of the constant assumption for f is that the geostrophic balance is not valid at the equator. However, a way to overcome this is to allow the Coriolis parameter to vary but also not to evaluate the $\frac{1}{f}$ term at the equator, where this term is not defined, as we would be dividing by zero.

The linear balance equation comes from differentiating the expressions for the geostrophic winds, which leads to the following Laplace equation for the geopotential;

$$\nabla^2\Phi = f_0\nabla^\psi = f_0\xi = \frac{1}{f_0}\left(\frac{\partial v_g}{\partial x} - \frac{\partial u_g}{\partial y}\right), \tag{17.2}$$

in Cartesian coordinates or

$$\nabla^2_{sp}\Phi = f_0\nabla^2_{sp}\psi = f_0\xi = f_0\left(\frac{1}{a\cos\theta}\frac{\partial v_g}{\partial \lambda} - \frac{1}{a}\frac{\cos\theta u_g}{\partial \theta}\right), \tag{17.3}$$

where ∇^2_{sp} is as defined in Chapter 12, and where ξ is the relative vorticity in spherical coordinates.

If we allow f to vary, then the **linear balance equation** in Cartesian coordinates is given by

$$\nabla^2\Phi = f\xi - u_g\frac{df}{dy} = \nabla\cdot(f\nabla\psi), \tag{17.4}$$

and in spherical coordinates by

$$\nabla^2\Phi = f\xi - \frac{u}{a}\frac{df}{d\theta} = \nabla_{sp}\cdot\left(f\nabla_{sp}\psi\right). \tag{17.5}$$

Linear balance is still used today, but not necessarily as a constraint of the initial conditions; it is more built into the covariance model or the control variable transform.

The next first-order balance that we present is referred to as **hydrostatic balance**. The hydrostatic balance equation can be expressed in terms of pressure gradients of the vertical wind, given by

$$\frac{dp}{dz} = -\rho g, \tag{17.6a}$$

$$w = -\rho g, \tag{17.6b}$$

where ρ is the density and g is the acceleration due to gravity. What the hydrostatic balance is implying is that the vertical motion upwards is equal to the downward motion of gravity, in other words there is a net zero motion in the vertical direction. This is true for most synoptic situation in the atmosphere, but even here there is some divergence away from this balance, but in a weak form. You can see hydrostatic balance when you are flying above the clouds and you look below and the cloud tops are all flat. However, when you are flying and you see the columns of clouds breaking through and growing upwards, this is where the motion is **not** in hydrostatic balance. As noted in [85], a consequence of the hydrostatic balance assumption is this type of model does not support sound waves.

The last balance that we consider here is the **thermal wind balance**. The **thermal wind** is defined as the vector difference in the geostrophic wind between two pressure levels, p_1 and p_0, where p_0 is closer to the surface. To obtain the thermal wind balance we consider the vertical derivative with respect to pressure of the geostrophic winds, which can be shown to be

$$\frac{\boldsymbol{u}_g}{\partial \ln p} = -\frac{R}{f}\mathbf{k} \times \nabla_p T, \tag{17.7}$$

where the interpretation of this balance, which is a function of geostrophic and hydrostatic balance, is that changes in the geostrophic winds between two isobaric levels p_1 and p_0 is directly related to the horizontal layer-mean temperature gradient.

So far the balance relationships presented are linear; however, most geophysical flows are not behaving linearly, and as such for certain scales we require expressions for nonlinear balance. A derivation of the nonlinear balance equation that relates geopotential to streamfunction can be found in [85], and we shall derive the nonlinear balance equation on the sphere later in this chapter when we consider higher-order balance equations. Thus, the nonlinear balance equation is given by

$$\nabla^2 \Phi = f\xi - \beta u_g - \nabla \cdot \left(\boldsymbol{u}_g \cdot \nabla \boldsymbol{u}_g\right), \tag{17.8}$$

where we see that the nonlinear balance term is allowing for more than rotational flow to be considered in balance.

Given these balances, we now consider different techniques to try to ensure that the numerical model does not excite spurious solutions.

17.1.2 Linear and Nonlinear Normal Mode Initialization

A very popular set of equations that are used for short and medium range weather forecasting are the primitive equations, but these equations admit high-frequency gravity wave solutions [82], as well as the slower-moving Rossby wave solutions. Although gravity waves are present in the atmosphere they are generally at smaller amplitude than the gravity waves that appear in primitive equations models. In numerical weather prediction it is the low-frequency part of the flow that is of interest and as such model gravity modes at worst can seriously compromise the forecast procedure.

According to Daley 1981,

> The gravity wave oscillations arise from initial imbalances between the wind and mass fields. These imbalances exist partially because the observed or analyzed mass and wind fields contain error and partially because the model equations do not exactly describe the atmosphere. Gravity wave oscillations can be controlled by the addition of time or space dissipation terms to the model equations. However, the principal way of suppressing gravity waves is by balancing the initial state through an **initialization** procedure.

It is stated in [82] that gravity waves generally require short computational time steps in the numerical model, and that they can seriously interfere with very short period forecasts, say less than 12 hours. They can also impair the precipitation and vertical motion calculations. Therefore it is advantageous if these spurious waves can be suppressed from the primitive equation model integrations.

One of the problems of some of the other procedures to damp out gravity modes is that they introduce artificial waves in the tropics. The starting point for normal mode initialization is to consider the equations of motion, thermodynamics, and continuity equation in pressure coordinates;

$$\frac{\partial \boldsymbol{u}}{\partial t} + \mathbf{k} \times f\boldsymbol{u} - \nabla \Phi = -\mathbf{k} \times \xi \boldsymbol{u} - \frac{1}{2}\nabla \boldsymbol{u} \cdot \boldsymbol{u} - \omega \frac{\partial \boldsymbol{u}}{\partial P}, \tag{17.9a}$$

$$\frac{\partial}{\partial t}\frac{\partial \Phi}{\partial P} + \frac{R_{\gamma^* \omega}}{P} = R - \boldsymbol{v} \cdot \nabla \frac{\partial \Phi}{\partial P} - \frac{R_{\gamma' \omega}}{P}, \tag{17.9b}$$

$$\frac{\partial \omega}{\partial P} + \nabla \cdot \boldsymbol{u} = 0, \tag{17.9c}$$

where ξ is the relative vorticity, $\xi = \mathbf{k} \times \nabla \times \boldsymbol{u}$, $\mathbf{k}$ is the unit vector in the vertical direction, $\gamma = -\frac{T}{\theta}\frac{\partial \theta}{\partial P} = \frac{RT}{C_p P} - \frac{\partial T}{\partial P}$, where θ is the potential temperature as defined in Chapter 10.

The next step is to introduce a linearization about the basic state of no flow with a basic state static stability γ^*, that is a function of the pressure, P, only. Next we attempt to separate the horizontal and vertical dependencies of the linearized equations, by eliminating the ω variable (see either Daley [82] or Daley [85] for more details). Thus for the horizontal wind fields u and v, and the geopotential height, Φ can be expressed as

$$\begin{pmatrix} u \\ v \\ \Phi \end{pmatrix} = \begin{pmatrix} \hat{u}(\lambda, \phi, t) \\ \hat{v}(\lambda, \phi, t) \\ \hat{\Phi}(\lambda, \phi, t) \end{pmatrix} Z(P), \tag{17.10}$$

where $Z(P)$ is the vertical structure, while $\hat{u}$, $\hat{v}$, and $\hat{\Phi}$ give the horizontal and temporal structure. Through the separation of variables technique, it can be shown that the linearized primitive equations are given by

$$\frac{\partial \hat{u}}{\partial t} - 2\Omega \hat{v} \sin\phi + \frac{1}{a\cos\phi}\frac{\partial \hat{\Phi}}{\partial \lambda} = 0, \tag{17.11a}$$

$$\frac{\partial \hat{v}}{\partial t} + 2\Omega \hat{u} \sin\phi + \frac{1}{a}\frac{\partial \hat{\Phi}}{\partial \phi} = 0, \tag{17.11b}$$

$$\frac{\partial \hat{\Phi}}{\partial t} + \frac{g\tilde{H}}{a\cos\phi}\left(\frac{\partial \hat{u}}{\partial \lambda} - \frac{\partial}{\partial \phi}\hat{v}\cos\phi\right) = 0, \tag{17.11c}$$

and a vertical structure equation which is a function of pressure only as

$$\frac{\partial}{\partial P}\frac{P}{R\gamma^*}\frac{\partial Z}{\partial P}+\frac{Z}{g\tilde{H}}=0, \tag{17.12}$$

where $\tilde{H}$ is the equivalent depth and $\left(g\tilde{H}\right)^{-1}$ is the separation constant.

We now assume an exponential behavior in time and longitude, such that

$$\begin{pmatrix}\hat{u}\\ \hat{v}\\ \hat{\Phi}\end{pmatrix}=\begin{pmatrix}\hat{u}^m\\ i\hat{v}\\ 2\Omega\hat{\Phi}^m\end{pmatrix}e^{im\lambda-2\Omega i\sigma t}, \tag{17.13}$$

where m is the zonal wave number, and σ is a non-dimensional frequency. Therefore, we can obtain expressions for coefficients of the exponential as

$$\sigma\hat{u}^m=-\hat{v}^m\sin\phi+\frac{m\hat{\Phi}^m}{a\cos\phi}, \tag{17.14a}$$

$$\sigma\hat{v}^m=-\hat{u}^m\sin\phi-\frac{1}{a}\frac{\partial\hat{\Phi}^m}{\partial\phi}, \tag{17.14b}$$

$$\sigma\hat{\Phi}^m=\frac{g\tilde{H}}{4\Omega^2a\cos\phi}\left(m\hat{u}^m+\frac{\partial}{\partial\phi}\hat{v}^m\cos\phi\right). \tag{17.14c}$$

The elimination of $\hat{u}^m$ and $\hat{v}^m$ between (17.14a) and (17.14b) with (17.14c) results in the **horizontal structure equation:**

$$H^m\left(\hat{\Phi}^m\right)+\frac{4\Omega^2a^2}{g\tilde{H}}\hat{\Phi}^m=0, \tag{17.15}$$

where

$$H^m\equiv\left(\frac{1}{\cos\phi}\frac{\partial}{\partial\phi}\left(\frac{\cos\phi}{\sigma^2-\sin^2\phi}-\frac{\partial}{\partial\phi}\right)+\frac{1}{\sigma^2-\sin^2\phi}\left(\frac{m}{\sigma}\frac{\sigma^2+\sin^2\phi}{\sigma^2-\sin^2\phi}-\frac{m^2}{\cos^2\phi}\right)\right), \tag{17.16}$$

is the **horizontal structure operator**.

Daley goes into more details about normal modes in [82] but in quite a wordy way. There is also a very good derivation of the normal mode initialization technique in [85], where according to [34], "*Daley devotes much of his book to the task of initialization*." However, there is a very clear mathematical expression for both linear and nonlinear normal mode initialization in [433] which we shall use to round out the theory of normal mode initialization.

Let $\boldsymbol{x}$ represent the vector of model variables, where these could be grid point values or spherical harmonic coefficients. Then the forecast model may be written in the form

$$\frac{\partial\boldsymbol{x}}{\partial t}=i\mathbf{A}\boldsymbol{x}+\mathcal{N}(\boldsymbol{x}), \tag{17.17}$$

where $\mathbf{A}$ is a constant coefficient matrix that represents the linear terms in the model, and the nonlinear terms are grouped together in $\mathcal{N}(\boldsymbol{x})$. It is assumed, for convenience, that the model variables have been scaled so that the matrix $\mathbf{A}$ is symmetric.

Therefore the model's normal modes are the eigenvectors of $\mathbf{A}$. The **normal mode decomposition** of $\mathbf{A}$ is simply

$$\mathbf{A} = \mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T, \tag{17.18}$$

where $\mathbf{E}$ is the matrix whose columns are the eigenvectors, $\boldsymbol{\xi}_n$, of $\mathbf{A}$, and $\boldsymbol{\Lambda}$ is the diagonal matrix of eigenvalues, λ_n, corresponding to the linear frequencies of the system in (17.17) with $\mathcal{N}(\boldsymbol{x})$ set to zero. As a result of assuming that $\mathbf{A}$ is symmetric, the columns of $\mathbf{E}$ are orthogonal, and as such $\mathbf{E}^T = \mathbf{E}^{-1}$, and the frequencies λ_n, are real valued $\forall n$.

The normal modes are classified as either slow modes, referred to as Rossby modes, or fast modes, referred to as gravity modes, and are the modes that we wish to control to prevent spurious waves being generated. Given this separation, it is possible to partition the $\mathbf{E}$ and $\boldsymbol{\Lambda}$ matrices as

$$\mathbf{E} = \left[\begin{array}{c|c} \mathbf{E}_R & \mathbf{E}_G \end{array}\right], \tag{17.19a}$$

$$\boldsymbol{\Lambda} = \left[\begin{array}{c|c} \boldsymbol{\Lambda}_R & \mathbf{0} \\ \hline \mathbf{0} & \boldsymbol{\Lambda}_G \end{array}\right], \tag{17.19b}$$

where the columns of $\mathbf{E}_R$ and $\mathbf{E}_G$ correspond to the slow and fast modes, respectively.

Through using normal mode decomposition, it is possible to transform between the vector $\boldsymbol{x}$ in physical space and a corresponding vector $\boldsymbol{c} = \mathbf{E}^T\boldsymbol{x}$ of normal mode coefficient. It is also possible to partition the $\boldsymbol{c}$ vector as

$$\boldsymbol{c} = \left[\begin{array}{c} \boldsymbol{c}_R \\ \hline \boldsymbol{c}_G \end{array}\right] = \left[\begin{array}{c} \mathbf{E}_R^T \\ \hline \mathbf{E}_G^T \end{array}\right]\boldsymbol{x}. \tag{17.20}$$

If we expand out the matrix-vector multiplication in (17.20), then we obtain

$$\boldsymbol{x} = \mathbf{E}\boldsymbol{c}\sum_n c_n\boldsymbol{\xi}_n, \tag{17.21}$$

which implies that $\boldsymbol{x}$ can be written as the sum of the Rossby wave and the gravity waves by

$$\boldsymbol{x} = \boldsymbol{x}_R + \boldsymbol{x}_G, \tag{17.22}$$

where we have the transforms

$$\boldsymbol{x}_R = \mathbf{E}_R\boldsymbol{c}_R = \mathbf{E}_R\mathbf{E}_R^T\boldsymbol{x}, \tag{17.23a}$$

$$\boldsymbol{x}_G = \mathbf{E}_G\boldsymbol{c}_G = \mathbf{E}_G\mathbf{E}_G^T\boldsymbol{x}. \tag{17.23b}$$

If we were to multiply (17.17) by $\mathbf{E}^T$, then we would obtain an equation for the time tendency of each of the normal mode coefficients as

$$\frac{dc_n}{dt} = i\lambda_n c_n + r_n, \tag{17.24}$$

where r_n is the projection of the nonlinear term $\mathcal{N}(\boldsymbol{x})$ on to the mode $\boldsymbol{\xi}_n$.

The point of normal mode initialization is that we wish to remove the fast modes that are responsible for the high-frequency oscillations as part of the initialization. A nonlinear normal mode initialization

could be achieved, according to [274], by assuming that the nonlinear term, r_n, varies quite slowly. If r_n is taken to be constant, then the solution of (17.24) is

$$c_n(t) = \left(c_n(0) - \frac{ir_n}{\lambda_n}\right) e^{i\lambda_n t} + \frac{ir_n}{\lambda_n}. \tag{17.25}$$

If we now set $c(0) = \frac{ir_n}{\lambda_n}$ for each of the fast modes, then the oscillatory term in (17.25) is eliminated. Since this relationship is nonlinear, as r_n is a function of c_n and so it has to be solved iteratively; in [274] this iterative scheme is given by

$$c_n^{(k+1)}(0) = \frac{ir_n^{(k)}}{\lambda_n}, \tag{17.26}$$

where (k) is the iteration number.

There are some complications to implement the initialization in (17.26) due to the need to explicitly evaluate the model's nonlinear terms before projecting them onto the fast modes, which may not be easy to do. To overcome this, Temperton refers to earlier papers where a trick is suggested, which is to run the numerical model forward one time step; therefore applying a forward time discrete scheme to (17.24) results in

$$\frac{c_n^{(k)}(\Delta t) - c_n^{(k)}(0)}{\Delta t} = i\lambda_n c_n^{(k)}(0) + r_n^{(k)}.$$

If we substitute (17.26) into the discretization above, then we obtain

$$c_n^{(k+1)}(0) = i\frac{c_n^{(k)}(\Delta_t) - c_n^{(k)}(0)}{\lambda_n \Delta t} + c_n^{(k)}(0),$$

which implies

$$\Delta c_n = i\frac{(\delta_t c_n)^{(k+1)}}{\lambda_n}, \tag{17.27}$$

where

$$\Delta c_n = c_n^{(k+1)}(0) - c_n^{(k)}(0),$$

is referred to as the change that has to be made to the coefficient of the fast mode, and

$$(\delta_t c_n)^{(k)} = \frac{c_n^{(k)}(\Delta t) - c_n^{(k)}(0)}{\Delta t},$$

is the tendency of the fast mode as evaluated through running the numerical model forward one time step.

In [433], Temperton provides a step-by-step description of the algorithm to implement this explicit nonlinear normal mode initialization procedure, which we summarize here:

- **Step 1:** Run the numerical model forward **one** time step to obtain a change in the state vector $\boldsymbol{x}$, denoted by $\delta_t \boldsymbol{x}$, which is the vector of tendencies of the model variable.
- **Step 2:** Compute $\delta_t \boldsymbol{c}_G = \mathbf{E}_G^T \delta_t \boldsymbol{x}$, which is equivalent to (17.20).

- **Step 3:** Compute $\Delta \boldsymbol{c}_G = i\boldsymbol{\Lambda}_G^{-1}\delta_t \boldsymbol{c}_G$, which is the vector of required changes to the coefficient of the fast mode (17.27).
- **Step 4:** Since only the fast modes need to be eliminated, which implies that $\Delta \boldsymbol{c}_R = \mathbf{0}$, then the required change to the state vector is $\Delta \boldsymbol{x} = \mathbf{E}_G \Delta \boldsymbol{c}_G$.

The four steps of the algorithm above can be written as a series of matrix-vector multiplications as

$$\Delta \boldsymbol{x} = i\mathbf{E}_G \boldsymbol{\Lambda}_G^{-1} \mathbf{E}_G^T \delta_t \boldsymbol{x}. \tag{17.28}$$

However, a drawback of this approach is that it requires explicit knowledge of each individual fast mode, $\boldsymbol{\xi}_n$, and its associated frequency, λ_n.

To overcome the requirement of having to know the fast modes in advance, Temperton suggests an implicit approach in [433] which we shall briefly summarize here.

Implicit nonlinear normal mode initialization

The starting point for implicit normal mode initialization is to assume that the matrix $\mathbf{A}$ is invertible, such that

$$\mathbf{A}^{-1} = \mathbf{E}\boldsymbol{\Lambda}^{-1}\mathbf{E}^T, \tag{17.29}$$

and to notice that the matrix on the right-hand side of (17.28) is similar to that of the expression in (17.29). The two expressions match exactly if we multiply (17.28) by $\mathbf{A} = \mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T$, such that

$$\mathbf{A}\Delta \boldsymbol{x} = i\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T \mathbf{E}_G \boldsymbol{\Lambda}_g^{-1} \mathbf{E}_G^T \delta_t \boldsymbol{x}. \tag{17.30}$$

As a result of the eigenvectors of $\mathbf{A}$ being orthogonal, then

$$\mathbf{E}^T \mathbf{E}_G = \left[\frac{\mathbf{0}}{\mathbf{I}}\right]. \tag{17.31}$$

Through the application of the matrix multiplication by blocks we have

$$\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T \mathbf{E}_G \boldsymbol{\Lambda}_G^{-1} = \mathbf{E}_G. \tag{17.32}$$

Substituting (17.32) into (17.30) results in the fundamental equation for implicit nonlinear normal model initialization, which is

$$\mathbf{A}\Delta \boldsymbol{x} = i\mathbf{E}_G \mathbf{E}_G^T \delta_t \boldsymbol{x}. \tag{17.33}$$

However, there is still one more step, and that is to separate $\delta_t \boldsymbol{x}$ into its two components

$$\delta_t \boldsymbol{x} = (\delta_t \boldsymbol{x})_R + (\delta_t \boldsymbol{x})_G\,. \tag{17.34}$$

A point to note here is that because we have the matrix $\mathbf{A}$ on the left-hand side in (17.33), then we have to solve a set of linear equations, but we already know $\mathbf{A}$. Therefore, it is stated in [433] that the two methods just presented are equivalent. Thus, the explicit nonlinear normal mode initialization can be implemented without knowing the normal modes, provided that:

1. given a vector $\delta_t \boldsymbol{x}$ of model tendencies, it can be separated into two components corresponding to the projection on to the slow and fast modes, respectively; and
2. the linear matrix vector system of the form $\mathbf{A}\Delta \boldsymbol{x} - \boldsymbol{y}$, can be solved.

However, while normal mode initialization is not used as much as it was, the requirement to stop the spurious fast motions in any form of geophysical modeling initialization is still a high priority. A drawback of the normal mode initialization is that it is applied to the analysis state from a form of data assimilation; this then makes the initialization state inconsistent with respect to the observations. The nonlinear normal mode initialization can be quite cumbersome to implement, and for limit area models it is less straightforward. However [186,273] introduced a digital filter technique to initialize forecasts that were produced with the High Resolution Limited Area Model (HIRLAM).

Digital filter

One way round creating inconsistent initial conditions to the observations from the normal mode initialization techniques is to introduce a new constraint term into the cost functions to ensure that the analysis state should not excite spurious gravity modes. This approach is referred to as adding a J_c term to the cost function, where this J_c term can take many different constraints that we wish to enforce on the analysis state. One form of "balancing constraint" is referred to as the **digital filter**.

The starting point of the derivation of the digital filter from [273] is to consider the filter of a continuous function first. Therefore let $f(t)$ be a continuous function that has both low- and high-frequency components, and we wish to filter/remove the higher frequencies. Thus the procedure to remove these frequencies is as follows:

1. Calculate the Fourier transform, denoted by $F(\omega)$ of $f(t)$.
2. Set the coefficients of the Fourier transform of the higher frequencies to zero.
3. Calculate the inverse transform.

To be able to achieve Step 2, we multiply the Fourier transformed function by a weighting function, $H(\omega)$, which could be a step function, defined as

$$H(\omega) = \begin{cases} 1, & |\omega| \leq |\omega_c| \\ 0, & |\omega| > |\omega_c| \end{cases}, \tag{17.35}$$

where ω_c is referred to as the cutoff frequency. The outcome of the three steps described above are equivalent to the convolution of $f(t)$ with $h(t) = \frac{\sin(\omega_c t)}{\pi t}$, which is the inverse Fourier transform of $H(\omega)$. This result follows from the convolution of Fourier transform presented in Chapter 12. Therefore, for us to be able to filter $f(t)$ we calculate

$$f^*(t) = (h * f)(t) = \int_{-\infty}^{\infty} h(\tau) f(t-\tau) d\tau. \tag{17.36}$$

However, in numerical modeling of different geophysical systems, we have that f is known only at discrete moments $t_n = n\Delta t$, which results in the sequence $\{\cdots, f_{-2}, f_{-1}, f_0, f_1, f_2, \cdots\}$. The sequence $\{f_n\}$ may be regarded as the Fourier coefficients of the function, $F(\theta)$, defined as

$$F(\theta) = \sum_{n=-\infty}^{\infty} f_n e^{-in\theta}, \tag{17.37}$$

where $\theta = \omega\Delta t$ is the **digital frequency** and $F(\theta)$ is periodic, such that $F(\theta) = F(\theta + 2\pi)$. Thus we are able to remove the high frequencies component of the sequence through multiplying $F(\theta)$ by a

function $H(\theta)$, given by

$$H(\theta) = \begin{cases} 1, & |\theta| \leq |\theta_c| \\ \\ 0, & |\theta| > |\theta_c| \end{cases}, \tag{17.38}$$

where the cutoff frequency $\theta_c = \omega_c \Delta t$, is assumed to fall in what is referred to as the Nyquist range $(-\pi, \pi)$, and $H(\theta)$ has period 2π. This enables us to expand the function, $H(\theta)$, as

$$H(\theta) = \sum_{n=-\infty}^{\infty} h_n e^{-in\theta}, \tag{17.39}$$

$$h_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} H(\theta) e^{-in\theta} d\theta. \tag{17.40}$$

The values of the h_n coefficients can be shown to be

$$h_n = \frac{\sin n\theta_c}{n\pi}. \tag{17.41}$$

We now denote the low frequency part of the sequence of $\{\mathbf{f_n}\}$ by $\left\{f_n^*\right\}$, from which all components greater than θ_c have been removed, which implies

$$H(\theta) \cdot F(\theta) = \sum_{n=-\infty}^{\infty} f_n^* e^{-in\theta}. \tag{17.42}$$

The convolution theorem for Fourier series implies that $H(\theta) \cdot F(\theta)$ is the transform of the convolution of the sequence of $\{h_n\}$ with $\{f_n\}$, which results in

$$f_n^* = (h * f)_n = \sum_{k=-\infty}^{\infty} h_k f_{n-k}. \tag{17.43}$$

The consequence of (17.43) is that it enables the filtering to be performed directly on the given sequence, and is also the discrete equivalent to (17.36).

However, in practice the summation must be truncated at some finite value of k, say N. This implies that an approximation to the low frequency components of the sequence, $\{f_n\}$, is given by

$$f_n^* = \sum_{k=-N}^{N} h_j f_{n-k}. \tag{17.44}$$

We have to be careful here as it is well known that the truncation of a Fourier series lead to Gibbs oscillations. In [273] the authors highlight that these oscillations need to be greatly reduced, otherwise what is the point of this theory if our goal is to remove spurious high frequency waves? In [273] the authors define a *window* (do not mistake this with the assimilation window) function. They suggest that the performance of the filter is improved if h_n is multiplied by what they refer to as the **Lanczos window**, which results in

$$w_n = \frac{\sin \frac{n\pi}{N+1}}{\frac{n\pi}{n+1}}. \tag{17.45}$$

The next feature to mention is the **transfer function**, $T(\theta)$, of a filter is defined as the function by which a pure sinusoidal oscillation is multiplied by, when subject to the filter. For symmetric coefficients, $h_k = h_{-k}$, the transfer function is real valued, implying that the phase is not altered by the filter. Therefore, if we have $f_n = e^{in\theta}$, then it is possible to write the transformed variable as $f_n^* = T(\theta) \cdot f_n$ and $T(\theta)$ is calculated by substituting f_n into (17.44), such that

$$T(\theta) = \sum_{k=-N}^{N} h_k e^{-ik\theta} = h_0 + 2\sum_{k=1}^{N} h_k \cos k\theta. \tag{17.46}$$

The digital filter is part of the signal processing theory from communications. It is stated that given a discrete function of time, $\{x_n\}$, then a **non-recursive** digital filter is defined by

$$y_n = \sum_{k=-N}^{N} a_k x_{n-k}, \tag{17.47}$$

where y_n is the output at time $n\Delta t$ depends of both the past and the future values of x_n, but not on other output values.

The **recursive** digital filter is defined by

$$y_n = \sum_{k=0}^{N} a_k x_{n-k} + \sum_{k=1}^{M} b_k y_{n-k}. \tag{17.48}$$

The output y_n at time $n\Delta t$ is (17.48) is dependent on past and present values of x_n, and also on the previous output values.

One way of implementing the digital filter is to integrate forward a set of initialized fields to a set time and obtain sums of the form

$$f_F^*(0) = \frac{1}{2} h_0 f_0 + \sum_{n=1}^{N} h_{-n} f_n. \tag{17.49}$$

The next step is to take the fields that have been stored for the same equivalent time we integrated forward but from before the analysis time, which yields

$$f_B^*(0) = \frac{1}{2} h_0 f_0 + \sum_{n=-1}^{-N} h_{-n} f_n. \tag{17.50}$$

Combining (17.49) and (17.50) yields the filtered data as

$$f^*(0) = f_F^*(0) + f_B^*(0). \tag{17.51}$$

17.2 Control Variable Transforms

An alternative approach that can be used, rather than initializing the analysis states to remove spurious gravity waves, is to try to build that structure into what are referred to as **control variable transforms**

(CVTs). The aim of CVTs is to select a set of variables that are less correlated than the model variables. We shall introduce different approach for this over the next three subsections, before the last four sections deal with techniques that have been suggested to bring the balance into the CVT, while still producing uncorrelated control variables.

17.2.1 Kinematic Approach

These sets of balance and unbalance decomposition that are presented in this section are referred to as **kinematic** in the sense that there is no real dynamical justification for the transform. However, it is assumed that the decomposition and the inverse transform create control variables that are uncorrelated. The transform that we refer to here is the **Helmholtz decomposition**.

The current method in use at the Met Office is outlined in Lorenc et al. [262]. There they describe how the control variables are defined in terms of a Helmholtz decomposition of the horizontal momentum together with a linear balance condition. The strategy is to define a (projective) transformation, $(u, v, p, \rho)) \mapsto \big(\psi, \chi, {}^{A}p\big)$, where the variables on the left are the two horizontal components of the wind field, the pressure, and the density, and the three variables on the right are a streamfunction, a velocity potential, and an "unbalanced" pressure. Then the horizontal wind field, $\boldsymbol{u}$, is decomposed into a balanced and an unbalanced variable through the pair of Poisson equations:

$$\xi \equiv \mathbf{k} \cdot \nabla \times \boldsymbol{u} = \nabla^2 \psi, \tag{17.52a}$$

$$\delta \equiv \nabla \cdot \boldsymbol{u} = \nabla^2 \chi, \tag{17.52b}$$

where ξ is the relative vorticity and δ is the divergence.

The unbalanced pressure, ${}^{A}p$, is defined by subtracting the pressure, p, calculated by solving the linear balance equation

$$\nabla \cdot (f\mathbf{k} \times \boldsymbol{u}) + \nabla \cdot (\gamma \nabla p) = 0 \tag{17.53}$$

for p given $\boldsymbol{u}$ and γ, from the total pressure. In (17.53), $\gamma = \rho^{-1}$ is the specific volume, which is usually approximated by $\gamma = \gamma(z)$ so that (17.53) is a constant-coefficient equation on each level, and f is the Coriolis parameter, defined by $f = 2\Omega \sin\theta$, where Ω is the Earth's rotation rate and θ is the angle of latitude, $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. The inverse transformation involves the Helmholtz decomposition

$$\boldsymbol{u} \equiv \mathbf{k} \times \nabla \psi + \nabla \chi \equiv \boldsymbol{u}_r + \boldsymbol{u}_d, \tag{17.54}$$

on each level, where $\boldsymbol{u}_r$ is the "balanced" rotational part of the wind field and $\boldsymbol{u}_d$ is the "unbalanced" divergent part. A "balanced" pressure is calculated from (17.53), and added to the unbalanced control variable ${}^{A}p$.

The reason for this type of decomposition is that we are trying to capture the slow Rossby mode in the control variable through the balanced streamfunction, and the two fast gravity modes through the unbalanced variables; velocity potential and unbalance pressure.

17.2.2 Spectral-Based CVT

ECMWF uses a wavelet formulation for its static component of the background error covariance modeling. The control variables at ECMWF are balanced vorticity, unbalanced divergence, temperature, and

surface pressure analyzed as a single parameter, mass, and the Holm moisture transform. The control variable is described in Derber and Bouttier [93].

The starting point for the background error covariance modeling is to define a change of variable, χ, such that $\boldsymbol{x} = L\chi$, with $\mathbf{L}^T\mathbf{B}^{-1}\mathbf{L} = \mathbf{I}$, where $\mathbf{L}$ is the transform between the uncorrelated control variables and the model variables. For the formulation in [93], it is assumed that temperature, surface pressure, (T, P_s), and divergence, δ, are partitioned into balanced and unbalanced components, allowing the $\mathbf{L}$ operator to be written as $\mathbf{L} = \mathbf{K}\mathbf{B}_u^{\frac{1}{2}}$ where $\mathbf{K}$ is a balance operator transforming the unbalanced variables into the model variables, and $\mathbf{B}_u$ is the symmetric square root of the background error covariance matrix $\mathbf{B}_u$ of the unbalanced variables. This matrix is assumed to be block diagonal with no correlations between the control variables and is defined by

$$\mathbf{B}_u = \begin{pmatrix} \mathbf{C}(\xi) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}(\delta_u) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}(T, p_s)_u & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{C}(q) \end{pmatrix}, \tag{17.55}$$

where $\mathbf{C}$ are covariance matrices.

The balance relationship, $\mathbf{K}$, is defined so that the balanced part of the divergence, along with the temperature and surface pressure, are given by the equations

$$\begin{aligned} \delta_b &= \mathbf{M}\xi, \\ (T, p_s) &= N\xi + P\delta_u, \end{aligned}$$

where $\mathbf{M}$ and $\mathbf{N}$ are the product of a horizontal balance operator, $\mathbf{H}_h$, which is a block-diagonal matrix transforming the spectral coefficients of vorticity independently at each vertical levels, into an intermediate variable P_b, which is a linearized mass variable, defined by

$$P_b = \beta_1(n, m)\,\xi(n, m+1) + \beta_2(n, m)\,\xi(m, m-1), \tag{17.56}$$

where n is the total wavenumber, m is the zonal wavenumber, and vertical balance operators $(\mathbf{M}_v, \mathbf{N}_v)$ such that $\mathbf{M} = \mathbf{M}_v\mathbf{H}_h$ and $\mathbf{N} = \mathbf{N}_v\mathbf{H}_h$.

We now assume that the covariance matrices, $\mathbf{C}$, are defined as

$$\mathbf{C} = \mathbf{S}^{-T}\mathbf{V}^{\frac{T}{2}}\mathbf{S}^T\mathbf{E}\mathbf{D}\mathbf{E}^T\mathbf{S}^{-1}\mathbf{V}^{\frac{1}{2}}\mathbf{S}, \tag{17.57}$$

where $\mathbf{S}$ represents the spectral to grid transformation, $\mathbf{V}$ is the diagonal matrix of variance of the unbalanced variables on the model Gaussian grid, $\mathbf{E}$ is the matrix of eigenvectors, and $\mathbf{D}$ is the diagonal matrix of eigenvalues of the vertical covariance matrices in spectral space of the unbalanced variables, normalized by the variances in grid space and assumed to vary with the total wavenumber. Therefore there is no correlation between different spectral coefficients, but a full autocovariance matrix is defined for each spectral coefficient. The resulting autocovariance model is non-separable, homogeneous, and isotropic in grid point space, where the grid point variances do not vary in the horizontal.

The algorithmic description is as follows:

First: The horizontal balance coefficients (β_1, β_2) are computed by a linear regression between the errors in vorticity and in linearized total mass, P_t as defined above, where $(\beta_1, \beta 2)$ are independent of

level. P_t is calculated from T and p_s through the linearized hydrostatic relationship at level l,

$$P_t(l) = \sum_{i=nlev}^{l} RT_i \Delta \log p_i + RT_{ref} \log p_S,$$

where T_{ref} is set to $270K$, $nlev$ is the number of model levels, and p_i is the set of pressures at the interfaces of the model layers.

Second: The vertical blocks of $\mathbf{M}_v$, one for each wavenumber, of this operator are computed by a linear regression between the spectral vertical profiles of coefficients of P_b and the divergence.

Third: The vertical blocks are computed for each total wavenumber as in step to but now for $\mathbf{N}_v$ but the linear regression is between vertical spectral profiles of P_b and $\delta_u = \delta - \mathbf{M}_v \xi$ to profiles of temperature and surface pressure: $[(T, p_s)]_n^m = \mathbf{N}_{v,n} [p_b]_n^m + P_n [\delta_u]_n^m$.

Fourth: The vertical autocovariances of the difference for the unbalanced variables $(\xi, \delta_u, (T, p_s)_u, q)$ are computed for each wavenumber.

17.2.3 Wavelet

The summary that we present here is from Fisher [121,123]. The basis of the wavelet formulation is to define a finite, non-orthogonal wavelet transform on the sphere in terms of radial basis functions, $\{\psi_j(r) : j = 1, 2, \ldots, K\}$ with the property

$$\sum_{j=1}^{K} \hat{\psi}_j^2(n) = 1, \quad \text{for } n = 0, 1, \ldots, n_{max}, \tag{17.58}$$

where $\hat{\psi}_j(n)$ is the nth coefficient of the Legendre transform of ψ, and where $\hat{\psi}_j(n) = 0$ for $n > n_{max}$.

We now consider a function f on the sphere whose spectral coefficients are zero for $n > n_{max}$. The wavelet coefficients are defined to be functions of f convolved with each of the radial basis functions as

$$f_j \psi_j \otimes f, \quad \text{for } j = 1, 2, \ldots, K. \tag{17.59}$$

The next stage is to consider the $(m, n)^{th}$ spectral component of the spherical harmonic expansion of the sum $\sum_{j=1}^{K} \psi_j \otimes f_j$ then

$$\begin{aligned} \sum_{j=1}^{K} \psi_j \otimes f_j &= \sum_{j=1}^{K} \hat{\psi}_j(n) \left(\hat{f}_j\right)_{m,n}, \\ &= \sum_{j=1}^{K} \hat{\psi}_j^2(n) \hat{f}_{m,n}, \\ &= \hat{f}_{m,n}. \end{aligned} \tag{17.60}$$

Finally, combining the information above enables the function f to be written as

$$f = \sum_j \psi_j \otimes f_j, \tag{17.61}$$

as given in [121]. More details, and an example of the wavelet formulation, can be found in [121,123].

We briefly summarize here how this is applied into a covariance model. The wavelet-based background error covariance model is derived from defining a control variable transform of the form

$$\boldsymbol{\chi} = \begin{pmatrix} \chi_1 \\ \chi_2 \\ \vdots \\ \chi_K \end{pmatrix}, \text{ where } \chi_j = \mathbf{C}_j^{-\frac{1}{2}}(\lambda,\theta)\left(\psi_j \otimes \boldsymbol{\Sigma}_b^{-\frac{1}{2}}\mathbf{T}(\boldsymbol{x}-\boldsymbol{x}_b)\right), \tag{17.62}$$

where $\boldsymbol{\Sigma}_b$ is the matrix representing the grid point variances of background error, and $\mathbf{T}$ is the matrix representing the balance operators as defined in the first part of this subsection.

In ECMWF's formulation of the CVTs, the background departures are defined as

$$\boldsymbol{x}-\boldsymbol{x}_b = \mathbf{T}^{-1}\boldsymbol{\Sigma}_b^{\frac{1}{2}}\sum_j \psi_j \otimes \left(\mathbf{C}_j^{\frac{1}{2}}(\lambda,\theta)\,\chi_j\right). \tag{17.63}$$

This is of the form where the $\mathbf{B}$ matrix can be factorized. It is stated in [121] that the expression above shows that the background departures at a given location are determined by the sum of convolutions of the functions $\mathbf{C}_j(\lambda,\theta)\,\chi_j$ with the functions ψ_j. It is stated that each convolution corresponds to a local averaging of nearby values, which comes about due to the property of the wavelet functions. By allowing the covariance matrix to vary with latitude and longitude, the covariances have spatial variation.

A more detailed description of many more different forms of CVTs, and the reasoning for the different choices at operational numerical weather prediction centers can be found in Bannister [22].

17.2.4 Nonlinear Balance on the Sphere

For us to be able to derive the nonlinear balance equation on the sphere, we have to introduce the shallow water equations in spherical coordinates. To arrive at the spherical version of the nonlinear balance equation we summarize the proof presented in [182].

Spherical shallow water equations

The vectorial version of the Cartesian form of the shallow water equations are given by

$$\frac{D\boldsymbol{u}}{Dt} + f\mathbf{k}\times\boldsymbol{u} = -g\nabla h.$$

Writing in component form gives, through using the expressions presented in Chapter 12, we have

$$\frac{\partial u}{\partial t} + \frac{u}{a\cos\theta}\frac{\partial u}{\partial \lambda} + \frac{v}{a}\frac{\partial u}{\partial \theta} - \frac{\tan\theta}{a}vu - fv = -\frac{g}{a\cos\theta}\frac{\partial h}{\partial \lambda}, \tag{17.64}$$

$$\frac{\partial v}{\partial t} + \frac{u}{a\cos\theta}\frac{\partial v}{\partial \lambda} + \frac{v}{a}\frac{\partial v}{\partial \theta} + \frac{\tan\theta}{a}u^2 + fu = -\frac{g}{a}\frac{\partial h}{\partial \theta}, \tag{17.65}$$

where a is the Earth's radius, $f = 2\Omega\sin\theta$, θ is the angle of latitude and has the values $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$, λ is the angle of longitude and has the value $\lambda \in [0, 2\pi]$ as defined before.

Using the information from Chapter 12 results in the spherical version of the continuity equation as

$$\frac{\partial h}{\partial t}+\frac{u}{a\cos\theta}\frac{\partial h}{\partial\lambda}+\frac{v}{a}\frac{\partial h}{\partial\theta}+\frac{h}{a\cos\theta}\left(\frac{\partial u}{\partial\lambda}+\frac{\partial(\cos\theta v)}{\partial\theta}\right)=0. \tag{17.66}$$

Therefore, (17.64), (17.65), and (17.66) are the spherical version of the shallow water equations.

Spherical nonlinear balance equation

In this subsection we derive the dimensional spherical version of the nonlinear balance equation. The starting point is to consider the spherical version of the equations of motion for the 3D primitive equations model. We make the assumption of homogeneity and ignore the vertical wind [473]. The remaining terms are similar to the spherical version of the shallow water equations, (17.64) and (17.65), but with geopotential gradients rather than height gradients.

We start by taking the divergence of the equations and ignoring the time derivative of the divergence. The reason for this is that the removal of this term "filters" the inertia-gravity waves [473]. The remaining terms are

$$\nabla\cdot((\boldsymbol{u}\cdot\nabla)\boldsymbol{u})+\nabla^2\Phi+\nabla\cdot(f\mathbf{k}\times\boldsymbol{u})=0. \tag{17.67}$$

Expanding the term $\nabla\cdot(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}$ using the spherical definitions in Chapter 12 gives

$$\begin{aligned}\nabla\cdot(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}=&\frac{1}{a^2\cos^2\theta}\left(\left(\frac{\partial u}{\partial\lambda}\right)^2+u\frac{\partial^2u}{\partial\lambda^2}\right)+\frac{1}{a^2\cos\phi}\left(2\frac{\partial v}{\partial\lambda}\frac{\partial u}{\partial\theta}+u\frac{\partial^2v}{\partial\theta\partial\lambda}+v\frac{\partial^2u}{\partial\lambda\partial\theta}\right)\\&-\frac{\tan\phi}{a^2\cos\theta}\left(v\frac{\partial u}{\partial\lambda}+2u\frac{\partial v}{\partial\lambda}\right)-\frac{\tan\theta}{a^2}v\frac{\partial v}{\partial\theta}+\left(\frac{1}{a^2\cos\theta}-\frac{\tan^2\theta}{a^2}\right)u^2\\&+\frac{1}{a^2}\left(\left(\frac{\partial v}{\partial\theta}\right)^2+v\frac{\partial^2v}{\partial\theta^2}+\frac{2\tan\theta}{a^2}u\frac{\partial u}{\partial\theta}\right).\end{aligned} \tag{17.68}$$

We now use the Helmholtz theorem for the wind field,

$$\boldsymbol{u}=\mathbf{k}\times\nabla\psi+\nabla\chi,$$

which in spherical coordinates, where we are only considering the balanced component, is

$$u=-\frac{1}{a}\frac{\partial\psi}{\partial\theta},\quad v=\frac{1}{a\cos\theta}\frac{\partial\psi}{\partial\lambda}, \tag{17.69}$$

and are the spherical definitions for the geostrophic winds from earlier. Substituting (17.69) into (17.68) results in

$$\begin{aligned}\nabla\cdot(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}=&\frac{1}{a^4\cos^2\theta}\left(2\left(\frac{\partial^2\psi}{\partial\theta\partial\lambda}\right)^2-2\frac{\partial^2\psi}{\partial\lambda^2}\frac{\partial^2\psi}{\partial\theta^2}+4\tan\theta\frac{\partial\psi}{\partial\lambda}\frac{\partial^2\psi}{\partial\theta\partial\lambda}\right.\\&\left.+\left(2\tan^2\theta+1\right)\left(\frac{\partial\psi}{\partial\theta}\right)^2\right)+\frac{1}{a^4}\left(\frac{\partial\psi}{\partial\theta}\right)^2+\frac{2\tan\theta}{a^4}\frac{\partial\psi}{\partial\theta}\frac{\partial^2\psi}{\partial\theta^2},\end{aligned} \tag{17.70}$$

where the third-order terms in (17.70) have canceled. We now consider the divergence of the Coriolis term, which is

$$\nabla \cdot (f\mathbf{k} \times \boldsymbol{u}) = \frac{1}{a\cos\theta}\frac{\partial}{\partial\lambda}(-fv) - \frac{\tan\theta}{a}(fu) + \frac{1}{a}\frac{\partial}{\partial\theta}(fu). \tag{17.71}$$

Substituting (17.69) for the wind field, and differentiating the Coriolis parameter with respect to θ, yields

$$\nabla \cdot (f\mathbf{k} \times \boldsymbol{u}) - \frac{1}{a^2\cos^2\theta} f\frac{\partial^2\psi}{\partial\lambda^2} + \frac{\tan\theta}{a^2} f\frac{\partial\psi}{\partial\theta} - \frac{\beta}{a^2}\frac{\partial\psi}{\partial\theta} - f\frac{\partial^2\psi}{\partial\theta^2}, \tag{17.72}$$

where

$$\beta = \frac{\partial f}{\partial\theta}.$$

Through collecting all the terms, and using the following notation

$$r = \frac{\partial^2\psi}{\partial\lambda^2}, \quad s = \frac{\partial^2\psi}{\partial\lambda\partial\theta}, \quad t = \frac{\partial^2\psi}{\partial\theta^2},$$

as used in [182], results in (17.67) becoming

$$\begin{aligned}\frac{2}{a^4\cos\theta}\left(rt - s^2\right) - \frac{4\tan\theta}{a^4\cos^2\theta}\psi_\lambda s + \left(\frac{f}{a^2} - \frac{2\tan\theta}{a^4}\psi_\theta\right)t + \frac{f}{a^2\cos^2\theta}r \\ = \left(\frac{f\tan\theta}{a^2} - \frac{\beta}{a}\right)\psi_\theta + \nabla^2\Phi + \frac{2\tan^2\theta}{a^4\cos^2\theta}\psi_\lambda^2 + \frac{1}{a^4}\left(\psi_\theta^2 + \frac{\psi_\lambda^2}{\cos^2\theta}\right).\end{aligned} \tag{17.73}$$

Therefore, (17.73) is a spherical version of a nonlinear balance equation for ψ, given Φ. Thus, the solution ψ, does not include inertia gravity waves. The differential equation for ψ is referred to as a **Monge-Ampère equation**.

The nonlinear balance equation that has been derived in this subsection is a nonlinear elliptical partial differential equation, and as such there is a mathematical condition that informs us if it is possible to find a solution to this differential equation. The condition of solvability is referred to as the **ellipticity condition**.

17.2.5 Ellipticity Conditions for Continuous PDEs

We start by considering the general form for a second-order partial differential equation

$$A(\theta,\lambda)\frac{\partial^2 S}{\partial\lambda^2} + B(\theta,\lambda)\frac{\partial^2 S}{\partial\theta\partial\lambda} + C(\theta,\lambda)\frac{\partial^2 S}{\partial\theta^2} + D(\theta,\lambda)\frac{\partial S}{\partial\theta} + E(\theta,\lambda)\frac{\partial S}{\partial\lambda} + F(\theta,\lambda)S = G(\theta,\lambda), \tag{17.74}$$

where the coefficients, $A,\ldots,G$ are functions of θ and λ and S is the solution to the equation. This can be classified as either a *hyperbolic*, *parabolic*, or *elliptic* equation [74,150].

Before we give the theorem for the existence of the unique solution for a general linear elliptic equation, we recall the definition for the types of partial differential equations and then specifically for the elliptic operator.

Definition 17.1. A partial differential equation of the form

$$AS_{\lambda\lambda} + BS_{\lambda\theta} + CS_{\theta\theta} + DS_{\lambda} + ES_{\theta} + FS = G, \tag{17.75}$$

where the coefficients, $A, \ldots, G$ are functions of θ and λ, is **hyperbolic** if $B^2 - 4AC > 0$, **parabolic** if $B^2 - 4AC = 0$ and **elliptic** if $B^2 - 4AC < 0$.

This enables the following definition for us to quantify if the operator is elliptic:

Definition 17.2. The differential operator

$$L[S] \equiv AS_{\lambda\lambda} + BS_{\lambda\theta} + CS_{\theta\theta}, \tag{17.76}$$

is elliptic if and only if $B^2 - 4AC < 0$.

We now give a specific version of a theorem from [74] that defines the existence and uniqueness of the solution to a homogeneous elliptic problem.

Theorem 17.3. *Given the elliptic operator, $L[S]$, then the differential equation*

$$L[S] + DS_{\lambda} + ES_{\theta} + FS = 0, \tag{17.77}$$

has one solution which has continuous derivatives up to second order in the interior of the domain and is continuous throughout the interior and the boundaries and assumes the prescribed boundary conditions values on the boundary.

We can classify the equations further with the following definition.

Definition 17.4. The partial differential equation, (17.74), is said to be **semi-linear** if A, B, and C are only functions of the independent variables, and **quasi-linear** if the same coefficients are functions of the independent variables and S, S_{θ}, or S_{λ}.

The inequality, $B^2 - 4AC < 0$ in Definition 17.2 is the **linear ellipticity condition** and is the condition that ensures that the differential equation has complex characteristics [74,150]. This condition is an important property of elliptic partial differential equations because if the coefficients are flow dependent, then there could be certain dynamical situations where a solution may not exist; however, if we violate the ellipticity condition, then this is saying that the partial differential equation changes type, either becoming a parabolic or hyperbolic partial differential equation.

Ellipticity condition for nonlinear partial differential equations

Nonlinear partial differential equations can also be classified as hyperbolic, parabolic, or elliptical; however, the condition changes from the linear expression in (17.75). We start with the general form for a nonlinear partial differential, which is

$$A + B\rho + 2C\nu + D\mu + E\left(\rho\mu - \nu^2\right) = 0, \tag{17.78}$$

where we have used the notation

$$p = \frac{\partial h}{\partial x}, \quad q = \frac{\partial h}{\partial y}, \quad \rho = \frac{\partial^2 h}{\partial x^2}, \quad \mu = \frac{\partial^2 h}{\partial y^2}, \quad \nu = \frac{\partial^2 h}{\partial x \partial y}.$$

The coefficients A, B, C, D, and E in (17.78) are given functions of (x, y, h, p, q). The ***nonlinear ellipticity condition*** [74] is given by

$$BD - C^2 - AE > 0. \tag{17.79}$$

We can see that the nonlinear ellipticity will collapse to the linear ellipticity condition in the situation where either A or E is equal to zero. The reason for presenting the two different forms of the ellipticity condition is due to the results that we present in Section 17.2.6, where we introduce a set of higher-order balance constraints that enable us to defined a balanced field consistent to a higher-order term in the asymptotic expansion of the wind fields about the Rossby number, but are the solution of nonlinear elliptical partial differential equations.

The importance of the higher-order balance is that it is stated in the normal mode initialization literature that initializing simply with the geostrophic wind does not damp out the initial gravity modes, and that the initial conditions need to be higher order than geostrophy. We shall now provide a summary of the derivation of the higher-order balance condition, which comes from [128].

17.2.6 Higher-Order Balance Conditions

The basis of the higher-order balance conditions we present in this section is the work of Rick Solomon in the 1980s [375–379], which then led to the development by McIntyre and Roulstone in 1999 and 2002 [287,288], which was then extended to a form of balance that could be applicable to incremental 4D VAR by the author [128]. The theory that enabled the derivations of the higher balance conditions was based upon Hamiltonian dynamics, which we summarize now.

Hamiltonian dynamics

We start by considering the shallow water equations where fluid is seen as a continuum, which is a continuous distribution of mass in space [378]. There are two ways of describing the continuum motion. First is the Eulerian approach, where the independent variables are the space coordinates, $\boldsymbol{x} = (x, y)$, and the time, t. The dependent variables are the height field, $h(x, y, t)$, and the velocities, $\boldsymbol{u}(x, y, t)$. Where the second way is the Lagrangian approach, where here the independent variables are a set of particle labels $\boldsymbol{a} = (a, b)$, and time τ. The dependent variables are the coordinates

$$x(a, b, \tau), \quad y(a, b, \tau), \tag{17.80}$$

at time τ, of the fluid parcel identified by (a, b). The particle labels vary continuously throughout the fluid, but the values of (a, b) on each fluid particle remain fixed as the fluid particle moves from place to place. Also the use of τ to denote time is to make clear that $\partial/\partial\tau$ means that (a, b) are being held fixed. For the Eulerian description $\partial/\partial t$ means that (x, y) are held fixed.

A way to view these descriptions is to think of a label space with coordinates (a, b) and a location space with coordinates (x, y). The fluid motion is a time-dependent mapping between these two spaces.

The derivatives with respect to Eulerian and Lagrangian coordinates are related by the chain rule

$$\frac{\partial F}{\partial \tau} = \frac{\partial F}{\partial t}\frac{\partial t}{\partial \tau} + \frac{\partial F}{\partial x}\frac{\partial x}{\partial \tau} + \frac{\partial F}{\partial y}\frac{\partial y}{\partial \tau}, \tag{17.81}$$

where F is a function of (x, y, t) or (a, b, τ). This leads to

$$\frac{\partial F}{\partial \tau} = \frac{\partial F}{\partial t} + u\frac{\partial F}{\partial x} + v\frac{\partial F}{\partial y} = \frac{\partial F}{\partial t} + \boldsymbol{u}\cdot\nabla F. \tag{17.82}$$

A detailed derivation of the properties above is given in [379].

With the basic description of fluid motions described here in terms of Lagrangian and Eulerian framework, we now show how these are used in derivation of motions using Hamilton's principle.

Hamilton's principle

Hamilton's principle states that the *action*

$$A \equiv \int_{t_1}^{t_2} L dt, \tag{17.83}$$

is stationary, where L is the *Lagrangian*, given by

$$L \equiv T - V, \tag{17.84}$$

where T is the kinetic energy and V is the potential energy. We briefly give two examples of how this is formulated. The first is for a system comprised of point masses while the second is a fluid continuum below.

For the first example we consider a system composed of N point-particles each with masses, m_i, $(i = 1 \text{ to } N)$, and locations, $\mathbf{x}_i(t)$. Now let $V(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ be the potential energy of the system. The kinetic energy is given by

$$T = \frac{1}{2} \sum_{i=1}^{N} m_i \frac{d\mathbf{x}_i}{dt} \cdot \frac{d\mathbf{x}_i}{dt}. \tag{17.85}$$

Hamilton's principle states that the first variation of the action, δA, satisfies

$$\delta A \equiv \delta \int_{t_1}^{t_2} \left(\frac{1}{2} \sum_{i=1}^{N} m_i \frac{d\mathbf{x}_i}{dt} \cdot \frac{d\mathbf{x}_i}{dt} - V \right) dt = 0, \tag{17.86}$$

for arbitrary, independent variations $\{\delta x_i(t), \delta y_i(t)\}$ that vanish at t_1 and t_2. Therefore we must have $\delta \mathbf{x}_i(t_1) = \delta \mathbf{x}_i(t_2) = 0$. Applying variational techniques, we obtain

$$0 = \int_{t_1}^{t_2} \left(-m_i \frac{d^2 \mathbf{x}_i}{dt^2} - \frac{dV}{d\mathbf{x}_i} \right) \cdot \delta \mathbf{x}_i dt. \tag{17.87}$$

As a result of the arbitrariness of the variations, the quantity inside the brackets must be zero. The result is Newton's second law.

For the second example we consider a barotropic fluid. The difference between the system of point masses and the fluid continuum is that the masses are distributed continuously in space in the continuum. Therefore, instead of a summation to represent the masses, we have

$$\iiint dtm = \iiint dadbdc, \tag{17.88}$$

as derived in [379]. The kinetic energy is given by

$$T = \frac{1}{2} \iiint dadbdc \frac{\partial \mathbf{x}}{\partial \tau} \cdot \frac{\partial \mathbf{x}}{\partial \tau}. \tag{17.89}$$

We assume that the potential energy arises from external and inter-particle forces that depend on the particle location $\mathbf{x}(a,b,c,\tau)$. The potential energy is

$$V = \iiint dadbdc\,(E(\alpha)+\phi(\mathbf{x})), \tag{17.90}$$

where α is the specific volume, given by

$$\alpha \equiv \frac{1}{\rho} = \frac{\partial(x,y,z)}{\partial(a,b,c)} = \begin{vmatrix} \frac{\partial x}{\partial a} & \frac{\partial x}{\partial b} & \frac{\partial x}{\partial c} \\ \frac{\partial y}{\partial a} & \frac{\partial y}{\partial b} & \frac{\partial y}{\partial c} \\ \frac{\partial z}{\partial a} & \frac{\partial z}{\partial b} & \frac{\partial z}{\partial c} \end{vmatrix} = \frac{\partial(\mathbf{x})}{\partial(\boldsymbol{a})},$$

ρ is the density, $E(\alpha)$ is the specific internal energy and is a function of α, and $\phi(\mathbf{x}(\boldsymbol{a},t))$ is the external potential, dependent on the fluid-particle locations.

Thus the *action* for this system is

$$\int d\tau\,(T-V) = \int d\tau \iiint d\boldsymbol{a}\left(\frac{1}{2}\frac{\partial \mathbf{x}}{\partial \tau}\cdot\frac{\partial \mathbf{x}}{\partial \tau} - E\left(\frac{\partial(\mathbf{x})}{\partial(\boldsymbol{a})}\right) - \phi(\mathbf{x}(\boldsymbol{a},\tau))\right), \tag{17.91}$$

which must be stationary with respect to the arbitrary variations $\delta\mathbf{x}(a,b,c,\tau)$, in the location of the fluid particles. A full derivation of the resulting equations can again be found in [379].

Hamiltonian form for the shallow water equations

In Salmon's 1983 and 1985 papers [375,376], he shows that Hamilton's principle for a mechanical system with N degrees of freedom can be written in the form

$$\delta\int d\tau\left(\sum_i p_i\frac{\partial q_i}{\partial \tau} - H(q_1,p_1,\ldots,q_N,p_N)\right) = 0, \tag{17.92}$$

where the variables q_i are the generalized coordinates, p_i are the corresponding momenta, H is the Hamiltonian and δ denotes the first variations at fixed time τ with respect to the arbitrary variations;

$$\delta q_i(\tau), \quad \delta p_i(\tau).$$

We consider the shallow water equations as a layer of inviscid homogeneous fluid. We use the definition for the height as the Jacobian of the mapping between the particle labels and the coordinates such that the conservation of mass is given by

$$dadb = \frac{dm}{\rho} = hdxdy, \tag{17.93}$$

where

$$h = \frac{\partial(a,b)}{\partial(x,y)} = \begin{vmatrix} \frac{\partial x}{\partial a} & \frac{\partial x}{\partial b} \\ \frac{\partial y}{\partial a} & \frac{\partial y}{\partial b} \end{vmatrix}. \tag{17.94}$$

In [375] the Lagrangian for the shallow water momentum equations is given by

$$L = \iint dadb\left((u-R)\frac{\partial x}{\partial \tau} + (v+P)\frac{\partial y}{\partial \tau}\right) - H, \tag{17.95}$$

where H is the *Hamiltonian* is given by

$$H = \frac{1}{2}\iint dadb\left(u^2 + v^2 + gh\right), \tag{17.96}$$

and the functions $R(x, y)$ and $P(x, y)$ represent the effect of rotation and have the property

$$\frac{\partial R}{\partial y} + \frac{\partial P}{\partial x} = f(x, y), \tag{17.97}$$

where f is the Coriolis parameter. Applying variations to x, y, u, and v results in the following Euler-Lagrange equations:

$$\delta x: \quad \frac{\partial u}{\partial \tau} - f\frac{\partial y}{\partial \tau} = -g\frac{\partial h}{\partial x}, \tag{17.98a}$$

$$\delta y: \quad \frac{\partial v}{\partial \tau} + f\frac{\partial x}{\partial \tau} = -g\frac{\partial h}{\partial y}, \tag{17.98b}$$

$$\delta u: \quad u = \frac{\partial x}{\partial \tau}, \tag{17.98c}$$

$$\delta v: \quad v = \frac{\partial y}{\partial \tau}. \tag{17.98d}$$

Substituting (17.98c) and (17.98d) into (17.98a) and (17.98b), then (17.98a) and (17.98b) are the shallow water momentum equations. The continuity equation arises from applying a variation to h in (17.95), given (17.96).

We now introduce an approximation that is made to the shallow water equations, which results in what are referred to as **semigeostrophic equations**.

Semi-geostrophic theory

We start by noticing that (17.98a) and (17.98b) can be written in terms of $(\ddot{x}, \ddot{y})$, where $(\dot{})$ represents $\frac{\partial}{\partial \tau}$, which enables (17.98a) and (17.98b) to be written as

$$\ddot{x} + g\frac{\partial h}{\partial x} - \dot{y}f = 0, \quad \ddot{y} + g\frac{\partial h}{\partial y} + \dot{x}f = 0. \tag{17.99}$$

The semi-geostrophic approximation is to replace the acceleration terms, $(\ddot{x}, \ddot{y})$, with the material derivatives of the geostrophic winds $\boldsymbol{u}_g = \begin{pmatrix} u_g & v_g \end{pmatrix}$, which leads to

$$\dot{u}_g + g\frac{\partial h}{\partial x} - \dot{y}f = 0, \quad \dot{v}_g + g\frac{\partial h}{\partial y} + \dot{x}f = 0. \tag{17.100}$$

The equations associated with each component of the vector in (17.100) are referred to as semi-geostrophic equations when they are combined with the continuity equation.

This system has the Hamiltonian

$$\mathbf{H} = V + \int_{\mathcal{D}} \frac{1}{2}|\boldsymbol{u}_g|^2 dm, \tag{17.101}$$

where $\mathcal{D}$ is the domain of interest and dm is the mass element, and

$$V = \int_{\mathcal{D}} \frac{1}{2} g h dm, \tag{17.102}$$

is the potential energy of the mass configuration. There is a conserved quantity like potential vorticity associated with this model, which is given by

$$Q_{sg} = \frac{1}{h}\left(f + \frac{\partial v_g}{\partial x} - \frac{\partial u_g}{\partial y} + \frac{1}{f}\frac{\partial\left(u_g, v_g\right)}{\partial\left(x, y\right)}\right). \tag{17.103}$$

Thus potential vorticity is materially invariant, i.e., $\frac{DQ_{sg}}{Dt} = 0$ [287].

17.2.7 Geostrophic Coordinates

Hoskins in his 1975 paper [180], showed that the semigeostrophic equations, (17.100), could be simplified through the following variable transform:

$$X = x + \frac{\partial \phi}{\partial x}, \quad Y = y + \frac{\partial \phi}{\partial y}, \tag{17.104}$$

where ϕ is defined by

$$\phi\left(x, y, t\right) = \frac{g}{f^2} h\left(x, y, t\right). \tag{17.105}$$

This choice of ϕ enables us to write the definition of geostrophic winds as

$$u_g = -f\frac{\partial \phi}{\partial y}, \quad v_g = f\frac{\partial \phi}{\partial x}. \tag{17.106}$$

This transformation, $\mathbf{x} \mapsto \boldsymbol{X}$, is referred to as the **geostrophic momentum transformation**. When f is assumed to be constant, then (X, Y) are referred to as the **momentum coordinates**, such that

$$\dot{X} = u_g, \quad \dot{Y} = v_g. \tag{17.107}$$

When $h\left(x, y, t\right)$, or correspondingly $\phi\left(x, y, t\right)$, is regarded as a known function of x, y with t fixed, then (17.104) specifies a transformation,

$$X = X\left(x, y, t\right), \quad Y = Y\left(x, y, t\right) \quad \left(t \text{ fixed}\right), \tag{17.108}$$

which at each time step is assumed to have an inverse of the form

$$x = x\left(X, Y, t\right), \quad y = y\left(X, Y, t\right). \tag{17.109}$$

We introduce a new variable dependent on the new coordinates,

$$\Phi\left(X, Y, t\right) = \phi + \frac{1}{2}\left(X - x\right)^2 + \left(Y - y\right)^2, \tag{17.110}$$

where Φ has the derivatives

$$\frac{\partial \Phi}{\partial X} = \frac{\partial \phi}{\partial x} = X - x, \quad \frac{\partial \Phi}{\partial y} = \frac{\partial \phi}{\partial y} = Y - y. \tag{17.111}$$

This enables us to write the material derivative of the geostrophic coordinates in terms of Φ as

$$\dot{X} = -f \frac{\partial \Phi}{\partial Y}, \quad \dot{Y} = f \frac{\partial \Phi}{\partial X}. \tag{17.112}$$

It is shown in [287] that

$$\begin{aligned} Q_{sg} &= \frac{1}{h}\left(f + \frac{\partial v^g}{\partial x} - \frac{\partial u^g}{\partial y} + \frac{1}{f}\frac{\partial \left(u^G, v^G\right)}{\partial (x, y)}\right), \\ &= \frac{g}{f\psi}\left(\frac{\partial X}{\partial x}\frac{\partial Y}{\partial y} - \frac{\partial X}{\partial y}\frac{\partial Y}{\partial x}\right), \\ &= \frac{g}{f^2\phi}\frac{\partial (X, Y)}{\partial (x, y)} > 0. \end{aligned} \tag{17.113}$$

Therefore, we are able to obtain the potential vorticity in terms of the new coordinates in a Jacobian form. Thus the semi-geostrophic evolution can be described in terms of this Jacobian.

In his series of papers in the 1980s, Salmon derived many different forms of balanced constraints through using the Hamiltonian functional, as well as linking them to the Hoskins' coordinate system. Given Salmon's work, McIntyre and Roulstone derived a general form of canonical coordinates that generate different balance conditions, through defining a general form for a balance wind field, given by

$$\boldsymbol{u}^c = \boldsymbol{u}_g + \frac{\alpha}{f}\boldsymbol{u}_g \cdot \nabla\left(\mathbf{k} \times \boldsymbol{u}_g\right), \tag{17.114}$$

where if $\alpha = 0$, then we obtain the geostrophic winds, if $\alpha = -\frac{1}{2}$, then we obtain the semi-geostrophic winds, and finally if $\alpha = 1$ then we obtain the Charney-Bolin balance, which is consistent with an asymptotic expansion of the wind field about the Rossby number to second order [406].

It is shown in [128] that it is possible to take the balanced wind field in (17.114) and derive elliptical Monge-Ampère equations on the sphere: first through expressing (17.114) in terms of spherical coordinates, and then forming either the relative vorticity or potential vorticity from (17.114), which results in the nonlinear balance equations in terms of Monge-Ampère equations. We refer the reader to [128] for the derivations of these balance equations; here we shall just present the spherical coordinate-based equations of the balanced wind field and the associated nonlinear balance equations.

First, the spherical coordinate version of (17.114) in component form is given by

$$u^c = u_g - \frac{\alpha}{f}\left(\frac{u_g}{a\cos\theta}\frac{\partial v_g}{\partial \lambda} + \frac{v_g}{a}\frac{\partial v_g}{\partial \theta} + \frac{\tan\theta}{a}u_g^2\right), \tag{17.115a}$$

$$v^c = v_g + \frac{\alpha}{f}\left(\frac{u_g}{a\cos\theta}\frac{\partial u_g}{\partial \lambda} + \frac{v_g}{a}\frac{\partial u_g}{\partial \theta} - \frac{\tan\theta}{a}u_g v_g\right). \tag{17.115b}$$

Assuming that the balance component is best described by the relative vorticity; we have to form ξ^c, and to do so we take the vertical component of the curl of (17.115a) and (17.115b). This gives

$$\xi^c \equiv \frac{1}{a\cos\theta}\left(\frac{\partial v^c}{\partial\lambda} - \frac{\partial}{\partial\theta}\left(\cos\theta u^c\right)\right), \tag{17.116}$$

which in component form is

$$\begin{aligned}\xi^c = \frac{1}{a\cos\theta}\Bigg(&\frac{\partial}{\partial\lambda}\left(v_g + \frac{\alpha}{f}\left(\frac{u_g}{a\cos\theta}\frac{\partial u_g}{\partial\lambda} + \frac{v_g}{a}\frac{\partial u_g}{\partial\theta} - \frac{\tan\theta}{a}u_g v_g\right)\right)\\ &+ \frac{\partial}{\partial\theta}\left(-\cos\theta u^c + \frac{\alpha}{f}\left(\frac{u_g}{a}\frac{\partial v_g}{\partial\lambda} + \frac{\cos\theta v_g}{a}\frac{\partial v_g}{\partial\theta} + \frac{\sin\theta}{a}u_g^2\right)\right)\Bigg).\end{aligned} \tag{17.117}$$

We now introduce the height version of the geostrophic winds in spherical coordinates. These are

$$u_g \equiv -\frac{g}{f}\frac{\partial h}{\partial\theta}, \quad v_g \equiv \frac{g}{fa\cos\theta}\frac{\partial h}{\partial\lambda}. \tag{17.118}$$

To derive the Monge-Ampère equation we substitute (17.118) into (17.117), which results in

$$\begin{aligned}\xi^c = \frac{g}{fa^2}\Bigg(&\frac{1}{\cos^2\theta}\frac{\partial^2 h}{\partial\lambda^2} + \frac{\partial^2 h}{\partial\theta^2} - \tan\theta\frac{\partial h}{\partial\theta}\Bigg) + \frac{2g^2\alpha}{f^3a^4\cos^2\theta}\Bigg(\left(\frac{\partial^2 h}{\partial\theta\partial\lambda}\right)^2\\ &-\frac{\partial^2 h}{\partial\lambda^2}\frac{\partial^2 h}{\partial\theta^2} + 2\tan\theta\frac{\partial h}{\partial\lambda}\frac{\partial^2 h}{\partial\theta\partial\lambda} + 2\tan^2\theta\left(\frac{\partial h}{\partial\lambda}\right)^2 + \sin\theta\cos\theta\frac{\partial h}{\partial\theta}\frac{\partial^2 h}{\partial\theta^2}\\ &+\frac{1}{2}\left(\left(\frac{\partial h}{\partial\lambda}\right)^2 + \cos^2\theta\left(\frac{\partial h}{\partial\theta}\right)^2\right)\Bigg).\end{aligned} \tag{17.119}$$

Note: all the third-order terms have canceled. To obtain this expression we assumed a constant f. This is referred to the f plane approximation [473], which is often used as a first stage of testing of new model variables.

If we consider the geostrophic subspace, $\alpha = 0$, then (17.119) simplifies to

$$\xi^c = \frac{g}{fa^2}\left(\frac{1}{\cos^2\theta}\frac{\partial^2 h}{\partial\lambda^2} + \frac{\partial^2 h}{\partial\theta^2} - \tan\theta\frac{\partial h}{\partial\theta}\right), \tag{17.120}$$

which is a spherical Poisson equation. If we take $\alpha = 1$, then the result is

$$\begin{aligned}\xi^c = \frac{g}{fa^2}\Bigg(&\frac{1}{\cos^2\theta}\frac{\partial^2 h}{\partial\lambda^2} + \frac{\partial^2 h}{\partial\theta^2} - \tan\theta\frac{\partial h}{\partial\theta}\Bigg) + \frac{2g^2}{f^3a^4\cos^2\theta}\Bigg(\left(\frac{\partial^2 h}{\partial\theta\partial\lambda}\right)^2\\ &-\frac{\partial^2 h}{\partial\lambda^2}\frac{\partial^2 h}{\partial\theta^2} + 2\tan\theta\frac{\partial h}{\partial\lambda}\frac{\partial^2 h}{\partial\theta\partial\lambda} + 2\tan^2\theta\left(\frac{\partial h}{\partial\lambda}\right)^2 + \sin\theta\cos\theta\frac{\partial h}{\partial\theta}\frac{\partial^2 h}{\partial\theta^2}\\ &+\frac{1}{2}\left(\left(\frac{\partial h}{\partial\lambda}\right)^2 + \cos^2\theta\left(\frac{\partial h}{\partial\theta}\right)^2\right)\Bigg).\end{aligned}$$

We now consider an extension to ξ^c to form the constrained potential vorticity, Q^c. We start from the definition

$$Q^c \equiv \frac{f+\xi^c}{h} \equiv \frac{\zeta^c}{h}, \tag{17.121}$$

where we would substitute the right-hand side of (17.119) into (17.121) for ξ^c, which results in

$$\begin{aligned} Q^c \equiv \frac{1}{h}\Bigg(f + \frac{g}{fa^2}\left(\frac{1}{\cos^2\theta}\frac{\partial^2 h}{\partial\lambda^2} + \frac{\partial^2 h}{\partial\theta^2} - \tan\theta\frac{\partial h}{\partial\theta}\right) + \frac{2g^2\alpha}{f^3a^4\cos^2\theta}\Bigg(\left(\frac{\partial^2 h}{\partial\theta\partial\lambda}\right)^2 \\ - \frac{\partial^2 h}{\partial\lambda^2}\frac{\partial^2 h}{\partial\theta^2} + 2\tan\theta\frac{\partial h}{\partial\lambda}\frac{\partial^2 h}{\partial\theta\partial\lambda} + 2\tan^2\theta\left(\frac{\partial h}{\partial\lambda}\right)^2 + \sin\theta\cos\theta\frac{\partial h}{\partial\theta}\frac{\partial^2 h}{\partial\theta^2} \\ + \frac{1}{2}\left(\left(\frac{\partial h}{\partial\lambda}\right)^2 + \cos^2\theta\left(\frac{\partial h}{\partial\theta}\right)^2\right)\Bigg)\Bigg). \end{aligned} \tag{17.122}$$

To use this in an incremental variational data assimilation scheme we would require a linear equation for the height increment. In the next section we introduce a linearization to (17.114) and derive a linearized Monge-Ampère equation for a height increment.

17.2.8 Linearization

In this section we introduce a linearization to (17.114) about a geostrophic base state. From the linearized version of the balanced wind field, we derive a linearized Monge-Ampère equation for the height increment.

Linearized balanced wind field

The nonlinear aspect of the balanced wind field, (17.114), arises from the term $\boldsymbol{u}_g \cdot \nabla\left(\mathbf{k}\times\boldsymbol{u}_g\right)$. To linearize this we introduce a base state for the height and consider increments to this. We start by expressing the height field, h, as $h = \bar{h} + h'$, where $\bar{h}$ is a base state height and h' is an increment. The geostrophic wind then becomes

$$\boldsymbol{u}_g = \bar{\boldsymbol{u}}_g + \boldsymbol{u}'_g, \tag{17.123}$$

where

$$u'_g \equiv -\frac{g}{af}\frac{\partial h'}{\partial\theta}, \quad v'_g \equiv \frac{g}{af\cos\theta}\frac{\partial h'}{\partial\lambda}, \tag{17.124}$$

$$\bar{u}_g \equiv -\frac{g}{af}\frac{\partial \bar{h}}{\partial\theta}, \quad \bar{v}_g \equiv \frac{g}{af\cos\theta}\frac{\partial \bar{h}}{\partial\lambda}. \tag{17.125}$$

Substituting (17.123) into (17.114) gives the increment to $\boldsymbol{u}^c$ as

$$\boldsymbol{u}^{c'} \equiv \boldsymbol{u}'_g + \frac{\alpha}{f}\left(\bar{\boldsymbol{u}}_g \cdot \nabla\left(\mathbf{k}\times\boldsymbol{u}'_g\right) + \boldsymbol{u}'_g \cdot \nabla\left(\mathbf{k}\times\bar{\boldsymbol{u}}_g\right)\right). \tag{17.126}$$

In component form, this is given by

$$u^{c'} \equiv u'_g - \frac{\alpha}{f}\left(\frac{u'_g}{a\cos\theta}\frac{\partial \overline{v}_g}{\partial \lambda} + \frac{\overline{u}_g}{a\cos\theta}\frac{\partial v'_g}{\partial \lambda} + \frac{v'_g}{a}\frac{\partial \overline{v}_g}{\partial \theta} + \frac{\overline{v}_g}{a}\frac{\partial v'_g}{\partial \theta} + 2\frac{\tan\theta}{a}u'_g\overline{u}_g\right), \tag{17.127a}$$

$$v^{c'} \equiv v'_g + \frac{\alpha}{f}\left(\frac{u'_g}{a\cos\theta}\frac{\partial \overline{u}_g}{\partial \lambda} + \frac{\overline{u}_g}{a\cos\theta}\frac{\partial u'_g}{\partial \lambda} + \frac{v'_g}{a}\frac{\partial \overline{u}_g}{\partial \theta} + \frac{\overline{v}_g}{a}\frac{\partial u'_g}{\partial \theta} - \frac{\tan\theta}{a}\left(u'_g\overline{v}_g + \overline{u}_g v'_g\right)\right). \tag{17.127b}$$

We can again construct a balance equation for the height, given either the constrained relative or potential vorticity. As before the reader is referred to [128] for the full derivation of the variable coefficient Poisson equations for $\alpha = 1$. Here we present the two linear balance equations:

$$\begin{aligned}\xi^{c'} &\equiv \mathbf{k}\cdot\nabla\times\boldsymbol{u}^{c'} \\ &\equiv \frac{g}{f}\nabla^2 h' - \frac{2\alpha g}{a^3 f^2\cos^2\theta}\left(2u_{g\lambda}h_{\theta\lambda} + \cos\theta v_{g\lambda}h_{\theta\theta} - u_{g\theta}h_{\lambda\lambda} + 2\tan\theta u_{g\lambda}h_\lambda - \cos\theta v_g h_\lambda\right. \\ &\quad \left. - 2\tan\theta\sin\theta v_g h_\lambda - 2\sin\theta v_g h_{\theta\lambda} + \cos^2\theta u_g h_\theta + \sin\theta\cos\theta\left(u_{g\theta}h_\theta + u_g h_{\theta\theta}\right)\right),\end{aligned} \tag{17.128}$$

where the subscripts λ and θ are referring to the derivative with respect to λ and θ, respectively.

For the potential vorticity-based approach we have a nonlinear expression in terms of the height field and as such we linearize the potential vorticity as

$$Q^{c'} = \frac{\mathbf{k}\cdot\nabla\times\boldsymbol{u}^{c'}}{\overline{h}} - \frac{\left(f + \mathbf{k}\cdot\nabla\times\overline{\boldsymbol{u}}^c\right)}{\overline{h}^2}h', \tag{17.129}$$

where

$$\begin{aligned}\bar{\xi}^c \equiv \mathbf{k}\cdot\nabla\times\overline{\boldsymbol{u}}^c &= \frac{1}{a\cos\theta}\left(\frac{\partial \overline{v}_g}{\partial\lambda} - \frac{\partial}{\partial\theta}\left(\cos\theta\overline{u}_g\right)\right) + \frac{2\alpha}{a^2 f\cos^2\theta}\left(2\left(\frac{\partial\overline{u}_g}{\partial\lambda}\right)^2\right. \\ &\quad + 2\cos\theta\frac{\partial\overline{v}_g}{\partial\lambda}\frac{\partial\overline{u}_g}{\partial\theta} - 4\sin\theta\overline{v}_g\frac{\partial\overline{u}_g}{\partial\lambda} + \left(2\sin^2\theta + \cos^2\theta\right)\overline{v}_g^2 \\ &\quad \left. + \cos^2\theta\overline{u}_g^2 + 2\sin\theta\cos\theta\overline{u}_g\frac{\partial\overline{u}_g}{\partial\theta}\right).\end{aligned} \tag{17.130}$$

The expressions above are quite complicated compared to simply solving a spherical Poisson equation, yet this is the price that we have to pay to introduce flow dependency into the balance decompositions. However, (17.128) and (17.129) are flow dependent but that they are also elliptic partial differential equations, as such they have associated ellipticity conditions. In [128] it is shown that the associated ellipticity conditions are

$$A(\theta,\lambda) = gf + 2g\alpha\frac{\partial\overline{u}_g}{\partial\theta}, \tag{17.131}$$

$$B(\theta,\lambda) = -4g\alpha\frac{\partial\overline{u}_g}{\partial\lambda} + \frac{4g\alpha\sin\theta}{a}\overline{v}_g, \tag{17.132}$$

$$C(\theta,\lambda) = gf\cos^2\theta - 2g\alpha\cos\theta\frac{\partial\overline{v}_g}{\partial\lambda} - \frac{2g\alpha\sin\theta\cos\theta}{a}\overline{u}_g. \tag{17.133}$$

The coefficients for the ellipticity condition, $B^2 - 4AC < 0$, are given by

$$B^2 = \alpha^2 \left(16g^2 \left(\frac{\partial \overline{u}_g}{\partial \lambda} \right)^2 - \frac{32g^2 \sin\theta}{a} \frac{\partial \overline{u}_g}{\partial \lambda} \overline{v}_g + \frac{16g^2 \sin^2\theta}{a^2} \overline{v}_g^2 \right), \tag{17.134}$$

$$\begin{aligned} 4AC = {} & 4g^2 f^2 \cos^2\theta - 8\alpha g^2 f \cos\theta \frac{\partial \overline{v}_g}{\partial \lambda} - \frac{\alpha 8g^2 f \sin\theta\cos\theta}{a} \overline{u}_g \\ & + 8\alpha g^2 f \cos^2\theta \frac{\partial \overline{u}_g}{\partial \theta} - 16\alpha^2 g^2 \cos\theta \frac{\partial \overline{v}_g}{\partial \lambda} \frac{\partial \overline{u}_g}{\partial \theta} - \frac{\alpha^2 16g^2 \sin\theta\cos\theta}{a} \frac{\partial \overline{u}_g}{\partial \theta} \overline{u}_g. \end{aligned} \tag{17.135}$$

Therefore, for the ellipticity condition to hold, and hence for there to be solutions, we require (17.134) $-$ (17.135) < 0. If we recall (17.119), we may write

$$\begin{aligned} \xi^c \equiv {} & \frac{1}{\cos\theta} \left(\frac{\partial v_g}{\partial \lambda} - \frac{\partial u_g}{\partial \theta} + \frac{\sin\theta}{a} u_g \right) + \frac{2\alpha}{f\cos^2\theta} \left(\left(\frac{\partial u_g}{\partial \lambda} \right)^2 + \cos\theta \frac{\partial v_g}{\partial \lambda} \frac{\partial u_g}{\partial \theta} \right. \\ & \left. - 2\frac{\sin\theta}{a} v_g \frac{\partial u_g}{\partial \lambda} + \frac{\sin^2\theta}{a^2} v_g^2 + \frac{\sin\theta\cos\theta}{a} u_g \frac{\partial u_g}{\partial \theta} + \frac{\cos^2\theta}{2a^2} \left(u_g^2 + v_g^2 \right) \right). \end{aligned} \tag{17.136}$$

Comparing (17.136) with (17.135), we see the ellipticity condition for (17.128) is

$$\alpha \cos^2\theta f \xi^c \left(\overline{u}_g, \overline{v}_g \right) < \frac{\cos^2\theta}{2a^2} \left(f^2 a^2 + \alpha^2 f \left(\overline{u}_g^2 + \overline{v}_g^2 \right) \right), \tag{17.137}$$

where $\xi^c \left(\overline{u}_g, \overline{v}_g \right)$ represents (17.136) evaluated with $\overline{u}_g, \overline{v}_g$ instead of u_g, v_g. This is therefore a bound on the choice of base state that may be used for the linearization. If we set $\alpha = 0$ in (17.137), then the condition for the interior of the domain, $\left(-\frac{\pi}{2}, \frac{\pi}{2} \right) \times [0, 2\pi)$, is

$$0 < f^2, \tag{17.138}$$

and is always satisfied, given the boundary condition, for a constant f.

An important feature to note here is that it can be shown that the ellipticity condition above is the same as the ellipticity condition for the nonlinear relative vorticity equation. Therefore, the significance of this result is to say that to ensure that the linearized partial differential equation is elliptic, the linearization state must satisfy the ellipticity condition for the nonlinear partial differential equation.

Following the same arguments, it can be shown that the ellipticity condition for (17.129) is the same as that of the ellipticity condition in (17.137); see [128] for more details.

The numerical study undertaken in [128] was with the Rossby-Haurwitz (RH) wave as described in [478]. In [128] the author consider three different forms of the RH waves that described three different flows in the shallow water equation model. In Figs. 17.1 and 17.2 we have plotted the ellipticity conditions for the initial conditions of the higher-order balance for these three flows and at 72 hours of time integration for the potential vorticity approach. We can see that there is not much change in the geostrophic cases, but that for the fast short motions, the ellipticity condition is quite flow dependent.

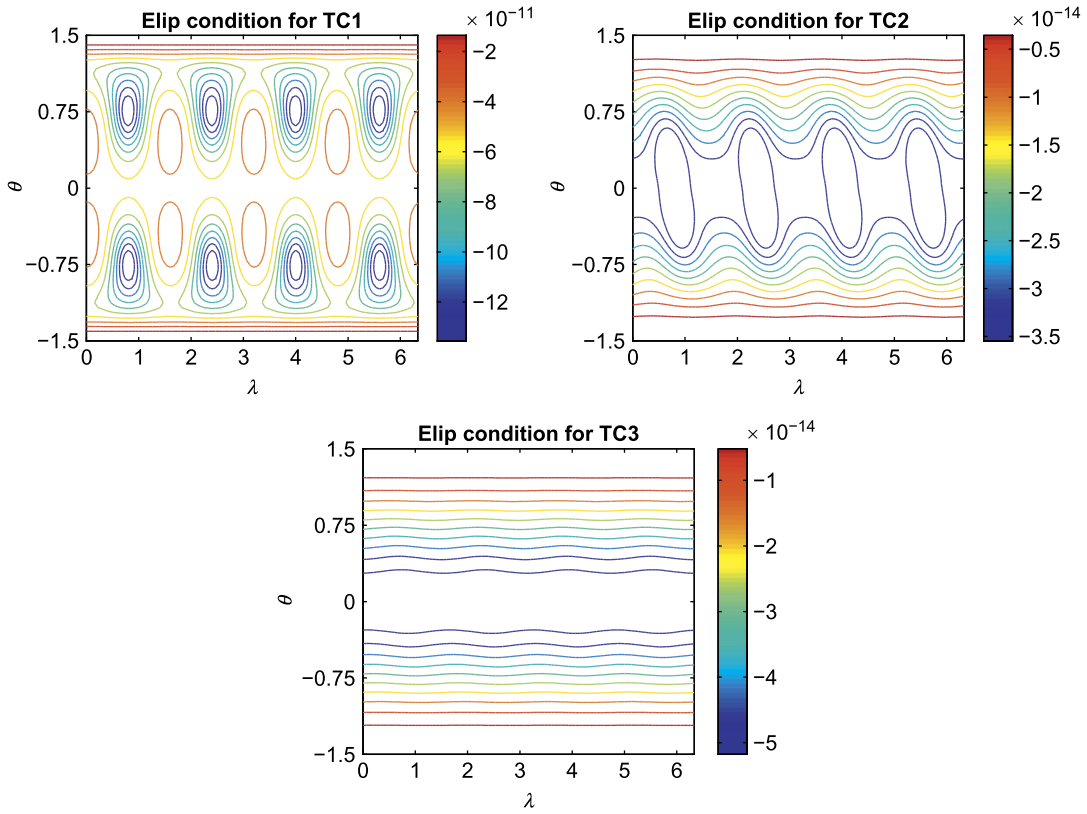


FIGURE 17.1

Plot of the initial ellipticity conditions of the three different Rossby-Hauwotz waves as described in [128] using potential vorticity as the balance variable.

17.3 Background Error Covariance Modeling

When the control variables have been decided upon, the next step is to design the covariance model for the background error covariance matrix. There is much debate about which method should be used, and here we present some of the choices that have been used in the past, and are still in use in some systems today. The models we present in this section are referred to as the **static component** as they do not change much throughout the year, where some seasonal and latitudinal variability may be used.

As we saw in the earlier part of this chapter, the need to stop the spurious gravity modes is imperative; an alternative approach to the normal mode initialization and the digital filters is to build the balances into the covariance model. One approach is to define a function to represent the height field error and use geostrophic balance, which implies that the error model for the wind fields is the derivative of this function. It is then possible to build the covariance model for the temperature field through

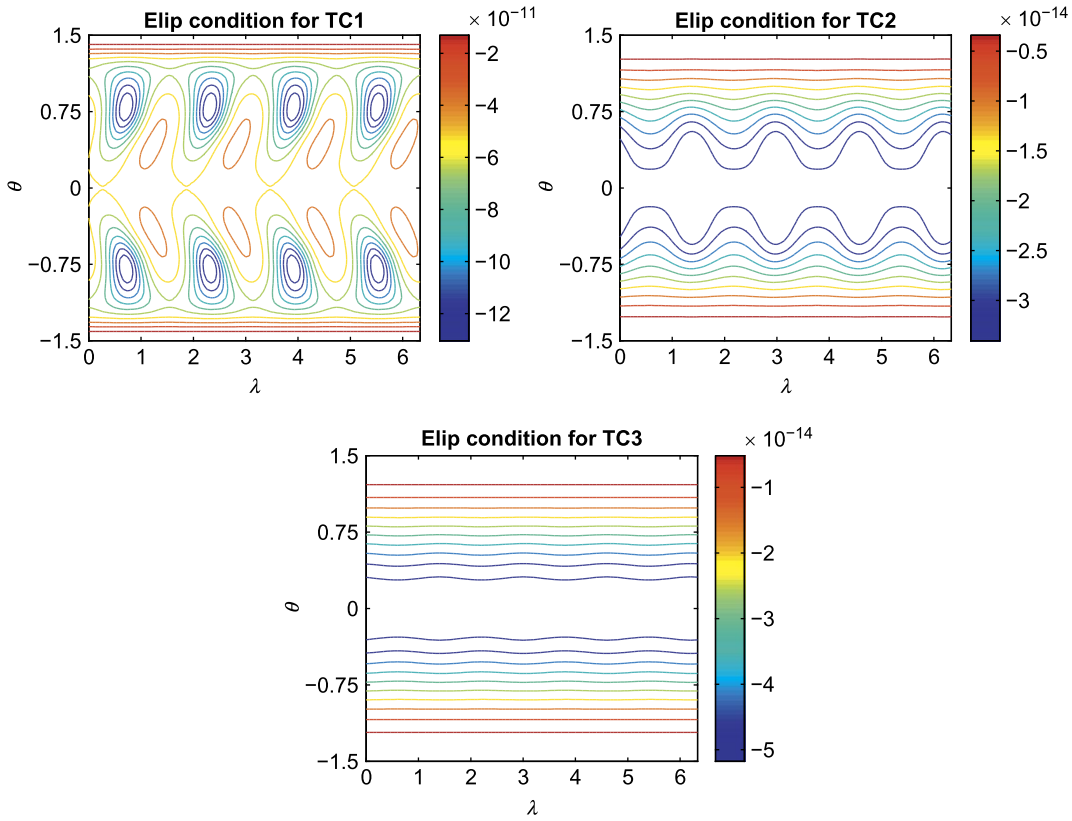


FIGURE 17.2

Plot of the ellipticity conditions at 72 hours of the three different Rossby-Hauwotz waves as described in [128] using potential vorticity as the balance variable.

the thermal wind balance, which is equivalent to integrating the height field's function. A very good detailed derivation of this approach can be found in [85].

17.3.1 Error Modeling Functions

In this subsection we consider different functions that have been used to determine the structure and the decorrelation lengths in different atmospheric and ocean operational numerical prediction centers over the last 35 years. We have seen some of the approaches in the optimal interpolation chapter as this is a requirement of these systems, but for the OI schemes we are only determining these lengthscales over a finite area, while we are solving globally in the VAR schemes.

Autoregressive models

The first auto regressive model that we present is the **second-order auto-regressive** model, or SOAR for short. This function is defined by

$$c_{n,m} = (1 + |\Delta\omega|)\, e^{-|\delta\omega|}, \tag{17.139}$$

where we are defining this model in the vertical, as such $\delta\omega = -\int_{z_m}^{z_n} L(z)^{-1}\, dz$, and $L(z)$ is a local vertical length scale that is a function of pressure.

The SOAR model in the horizontal direction is defined as

$$c_b^h(s_{nm}, L_{nm}) = \left(1 + \frac{s_{nm}}{L_{nm}}\right) e^{-\frac{s_{nm}}{L_{nm}}}, \tag{17.140}$$

where s_{nm} is some form of distance measure between points on the type of grid being considered, and L_{nm} is the decorrelation length in that specific horizontal direction. A **third-order auto-regressive**, or **TOAR**, as stated in [153], can be defined as

$$F(c, r) = (1 + \alpha)^{-1} \left(f(c, r) + \alpha f\left(\frac{c}{N}, r\right)\right), \tag{17.141}$$

where

$$f(c, r) = \left(1 + cr + \frac{c^2 r^2}{c}\right) e^{-cr}, \tag{17.142}$$

and

$$L_c = \sqrt{\left(\frac{3(1+\alpha)}{1 + \frac{\alpha}{N^2}}\right)} c^{-1}, \tag{17.143}$$

where L_c is the correlation length scale, with c inversely related to this length scale. It is also stated in [153] that is possible to fit the covariance model to the wind fields from the function for the covariance for the geopotential height in (17.141) as

$$F'(c, r) = \left(1 + \frac{\alpha}{N^2}\right)^{-1} \left(f'(c, r) + \frac{\alpha}{N^2} f'\left(\frac{c}{N}, r\right)\right), \tag{17.144a}$$

$$f'(c, r) = (1 + cr)\, e^{-cr}, \tag{17.144b}$$

where in [153] the values for α is 0.2 and for N is 3.

Gaussian

An alternative to the auto-regressive models in the vertical is to fit a Gaussian for the decorrelation, which is given by

$$c_{n,m} = e^{-\frac{(\Delta\omega)^2}{2}}. \tag{17.145}$$

The reason why people may not use the Gaussian, and prefer the SOAR approach, is due to the slope, effectively the dropoff of the correlations as a function of distance. We can see from Fig. 17.3 that there

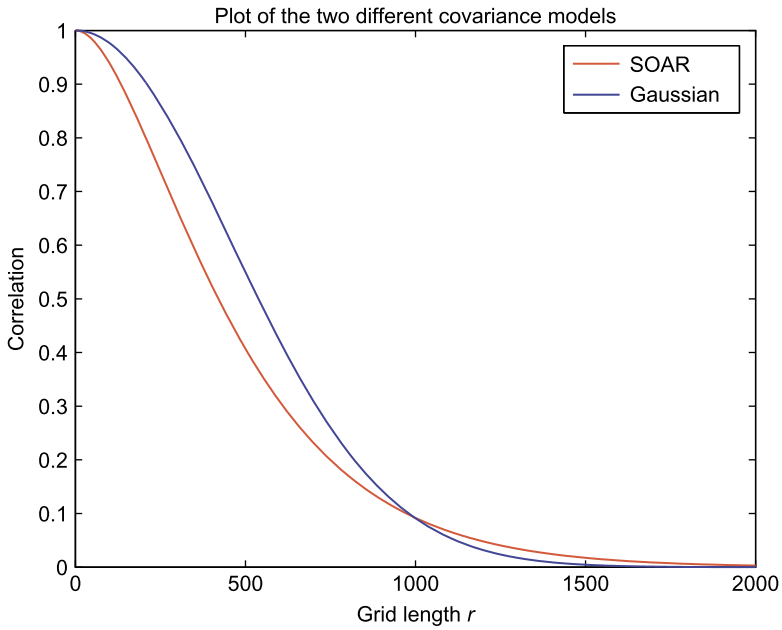


FIGURE 17.3

Plot of the SOAR and the Gaussian correlation functions.

is a subtle difference between the two approaches, where the Gaussian allows for a longer dropoff, whereas the SOAR approximation does not allow as far correlations.

The horizontal version of the Gaussian model for the correlation is given by

$$c_b^h(s_{nm}, L_{nm}) = e^{-\left(\frac{s_{nm}}{L_{nm}}\right)^2}. \tag{17.146}$$

Compact spline

The covariance model here comes from [152], where this a function of compact support that goes identically to zero with its first two derivatives at some finite distance, which is denoted as $2c$. If we let $c = \left(\frac{10}{3}\right)^{\frac{1}{2}}$ and $r = \frac{s_{nm}}{cL_{nm}}$, then the correlation model in the horizontal is given by

$$c_b^h = -\frac{r^5}{4} + \frac{r^4}{2} + \frac{5r^4}{8} - \frac{5r^2}{3} + 1, \qquad 0 \le r \le 1, \tag{17.147a}$$

$$= \frac{r^5}{12} - \frac{r^4}{2} + \frac{5r^3}{8} + \frac{5r^2}{3} - 5r + 4 - \frac{2}{3r}, \qquad 1 < r \le 2, \tag{17.147b}$$

$$= 0, \qquad r > 2. \tag{17.147c}$$

Gaussian localization function with compact support

In [152] there are a series of different decorrelation length models that are presented, along with very rigorous mathematical proofs of theorems proving the support the different scheme provide. The reader

is referred to [152] for these details, but one of the often used schemes from there is the 5th order compact support polynomial given by

$$C(d,c) = \begin{cases} 1 - \frac{20}{3}\left(\frac{d}{c}\right)^2 + 5\left(\frac{d}{c}\right)^3 + 8\left(\frac{d}{c}\right)^4 - 8\left(\frac{d}{c}\right)^5 & 0 \le d \le \frac{c}{2}, \\ \frac{8}{3}\left(\frac{d}{c}\right)^5 - 8\left(\frac{d}{c}\right)^4 + 5\left(\frac{d}{c}\right)^3 + \frac{20}{3}\left(\frac{d}{c}\right)^2 - 10\left(\frac{d}{c}\right) + 4 - \frac{1}{3}\left(\frac{d}{c}\right)^{-1} & \frac{1}{2}c \le d \le c, \\ 0 & c \le d, \end{cases} \tag{17.148}$$

where d is the distance between observations and analysis grid, and c is referred to as the cutoff distance.

17.3.2 Determining Variances and Decorrelation Lengths

Having determined which model to describe the background errors covariances, we need to determine the values for the variances and decorrelation length scales. In this subsection we present two different approaches that have been used to determine these estimates.

National Meteorological Center approach

One of the most widely used methods to determine the variance and the decorrelation lengths is referred to as the National Meteorological Center (NMC) method; the acronym comes from the NMC, which is now the National Center for Environmental Prediction (NCEP). The NMC method first appeared in the 1992 paper by David Parrish and John Derber [328], and is based on the idea that the differences between two lagged forecast that are valid at the same time give an estimation of the background error variance and covariance. The idea of the NMC method is to fit the correlation function of choice and determine the correlation length scale that the climatology of 24-hour lagged forecast differences; usually the difference is between a 24- and 48-hour forecast valid at the same time. One of the assumptions of this scheme is that the error growth is linear in this window for synoptic weather errors. Note that in other applications shorter or longer lag times may be a better model for those dynamics.

Recently there has been work at the National Aeronautic and Space Agency (NASA) that shows that the estimates from the NMC-based approach need to be rescaled [105]; the abstract for this paper states:

> The NMC method has proven utility for prescribing approximate background-error covariances required by variational data assimilation systems. Here, untuned NMC method estimates are compared with explicitly determined error covariances produced within an OSSE context by exploiting availability of the true simulated states. Such a comparison provides insights into what kind of rescaling is required to render the NMC method estimates usable. It is shown that rescaling of variances and directional correlation lengths depends greatly on both pressure and latitude. In particular, some scaling coefficients appropriate in the Tropics are the reciprocal of those in the Extratropics. Also, the degree of dynamic balance is grossly overestimated by the NMC method. These results agree with previous examinations of the NMC method which used ensembles as an alternative for estimating background-error statistics.

We refer the reader to [105] for a more detail explanation of the rescaling that is needed when using the NMC method.

Observation-based approach

Before the introduction of the NMC method in [328], a widely used technique to estimate the background error characteristics was a method introduced by **Hollingsworth and Lönnberg** in 1986 [179,256]. The basis of this scheme is to assume that the background errors are independent of observation errors, and along with spatially uncorrelated observations errors. Given these assumptions, the innovation are defined by $\mathbf{d}_i = \boldsymbol{y}_i - \boldsymbol{h}(\boldsymbol{x}_b)$ for the ith observation. If we denote the background error as $\boldsymbol{\varepsilon}_b$ and the observational error as $\boldsymbol{\varepsilon}_o$, where we assume that there is no representative error, then

$$\mathbf{d}_i = \boldsymbol{\varepsilon}_o + \boldsymbol{h}(\boldsymbol{\varepsilon}_b). \tag{17.149}$$

Given the expression in (17.149), then

$$Var[\mathbf{d}_i] = Var[\boldsymbol{\varepsilon_o}] + Var[\boldsymbol{\varepsilon_b}], \tag{17.150a}$$

$$CoV[\mathbf{d}_i, \mathbf{d}_k] = Cov[\boldsymbol{h}_i(\boldsymbol{\varepsilon}_b), \boldsymbol{h}_k(\boldsymbol{\varepsilon}_b)], \tag{17.150b}$$

where the observations at points i and k are not co-located. The significance of (17.150a) and (17.150b) is that they are saying that the total observational variance, as determined at a point i, comprises of the measurement error variance and the variance of the observation operator acting on the background error. The next important property of (17.36) is that the covariance between innovation vectors at different locations is only a function of the background error. We should note that the estimation of the variances and the covariances come about through taking samples between multiple observations at a set distance apart and then plotting, or analyzing, these differences that are organized by observation separation distance. We have a copy of figure 6 from [179] for temperature correlations in Fig. 17.4.

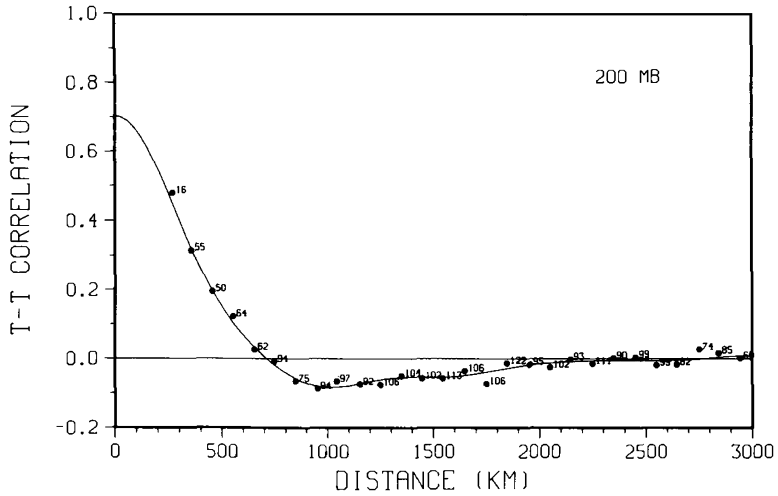


FIGURE 17.4

Copy of figure 6 from Hollingsworth, A. and Lönnberg, P. (1986), The statistical structure of short-range forecast errors as determined from radiosonde data. Part I: The wind field. Tellus A, 38A: 111-136. https://doi.org/10.3402/tellusa.v38i2.11707, diagnosing the correlation lengths for temperature.

17.4 Preconditioning

Recalling the 3D VAR cost function for Gaussian background and observational error

$$J(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_b)^T \mathbf{B}^{-1}(\boldsymbol{x}-\boldsymbol{x}_b) + \frac{1}{2}(\boldsymbol{y}-\boldsymbol{h}(\boldsymbol{x}))^T \mathbf{R}^{-1}(\boldsymbol{y}-\boldsymbol{h}(\boldsymbol{x})),$$

and as mentioned for certain applications, the dimensions of the matrices are too large to be stored, or even explicitly calculated. Another problem is that the derivative-free algorithms for the minimization of the cost function are too slow, which is due to the fact that each function evaluation provides very limited information about the shape of the cost function and in which direction the minimum may be. Therefore, we are going to require algorithms that utilize the gradient of the cost function. We shall go into more detail about some specific algorithms in the next section.

However, from a motivation point of view for why we wish to precondition the problem, we consider the simplest gradient-based minimization algorithm which is the **steepest descent method**. This approach is based on the following four steps:

1. Define a **descent direction**: $\mathbf{d}_k = -\nabla J(\boldsymbol{x}_k)$.
2. Find a **step**, α_k, which is referred to as a **line search** for which $J(\boldsymbol{x}+\alpha\mathbf{d}_k) < J(\boldsymbol{x}_k)$.
3. Set $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha\mathbf{d}_k$.
4. Repeat until the gradient is sufficiently small.

The steepest descent approach works well on very well conditioned problems where the iso-surfaces of the cost function are nearly spherical. However, it is often the case that these iso-surfaces are ellipsoidal initially and so at this point the steepest descent is not efficient. In Fig. 17.5 is a schematic to illustrate the step directions the steepest descent algorithms could take for each iteration.

Another downside to the steepest descent algorithm is that it is inefficient as does not use information about the **curvature** of the cost function. The curvature information comes from the Hessian of the cost function. The algorithms that use the curvature information are referred to as **Newton or Newton-Raphson methods**.

To ascertain the degree of sphericity of the cost function, we require the eigenvalues of the Hessian matrix as each eigenvalue of this matrix represents the curvature in the direction of the corresponding eigenvector. Another important feature of the eigenvalues of the Hessian matrix is that they can inform

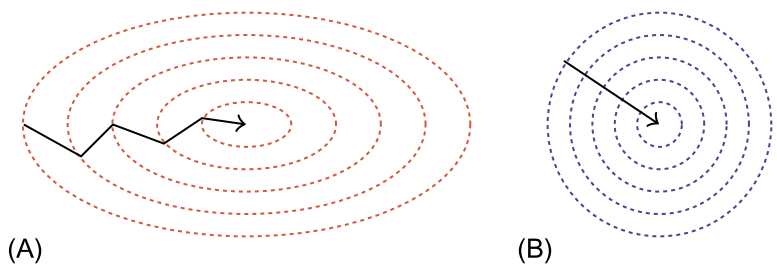


FIGURE 17.5

Schematic of the steepest decent method for (A) unpreconditioned and (B) preconditioned.

us of the convergence rate due to condition number of the matrix;

$$\kappa = \frac{\lambda_{max}}{\lambda_{min}}, \tag{17.151}$$

where λ are the eigenvalues of the Hessian matrix. Therefore, the convergence rate can be accelerated by reducing the condition number of the Hessian matrix. The Hessian of the 3D VAR cost function is

$$\mathbf{J}'' = \mathbf{B}^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}, \tag{17.152}$$

and it is this matrix that we wish to change in the data assimilation application.

An approach to try to speed up the convergence of the minimization algorithm is to introduce a change of variable $\boldsymbol{X} = \mathbf{L}^{-1} \delta \boldsymbol{x}$ for the incremental variational formulation, which implies that $\delta \boldsymbol{x} = \mathbf{L}\mathbf{J}''$. Many different operational numerical weather prediction centers quite often approximate the $\mathbf{L}$ matrix by $\mathbf{B}^{\frac{1}{2}}$; however, each center has different formulations for $\mathbf{B}^{\frac{1}{2}}$. We saw how the ECMWF approximates its $\mathbf{B}$ matrix through a series of balanced and unbalanced decompositions and linear regressions. The Met. Office used a different approach due to their global model at the time being in grid space. This approach is used in the Weather, Research, and Forecasting (WRF) Global Statistical Interpolation (GSI) system. We shall present the Met. Office's T and U transforms in Chapter 20, in the description of the hybrid system.

If we now substitute $\mathbf{L} = \mathbf{B}^{\frac{1}{2}}$ into the incremental version of 3D VAR cost function, then we obtain a new cost function that is in terms of the transformed variable $\boldsymbol{X}$ as

$$J(\boldsymbol{X}) = \frac{1}{2} \boldsymbol{X}^T \boldsymbol{X} + \frac{1}{2} (\mathbf{d} - \mathbf{H}\mathbf{L}\boldsymbol{X})^T \mathbf{R} (\mathbf{d} - \mathbf{H}\mathbf{L}\boldsymbol{X}), \tag{17.153}$$

where $\mathbf{d}$ is the innovation vector as we have seen before. It can easily be shown that the Hessian matrix of (17.153) is

$$\mathbf{J}''_X = \mathbf{I} + \mathbf{L}^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H}\mathbf{L}. \tag{17.154}$$

Exercise 17.5. *Prove that* (17.154) *is the Hessian matrix for* (17.153).

An important feature to note about the Hessian matrix in (17.154) is that it contains the identity matrix, $\mathbf{I}$. This means that all of the eigenvalues here are greater than or equal to 1, which implies that there are no small eigenvalues that could negatively impact the conditioning of the problem.

However, using a square root approximation, recalling that the square roots of matrices are non-unique, is still not the optimal preconditioner. If we recall the incremental version of VAR, then it is possible to rewrite the cost function as

$$J(\delta \boldsymbol{x}) = \frac{1}{2} \delta \boldsymbol{x}^T \mathbf{J}'' \delta \boldsymbol{x} + \boldsymbol{g}^t \delta \boldsymbol{x} + c. \tag{17.155}$$

If we apply a preconditioner to (17.155) then we obtain

$$J(\boldsymbol{X}) = \frac{1}{2} \boldsymbol{X}^T \mathbf{L}\mathbf{J}''\mathbf{L}\boldsymbol{X} + \boldsymbol{g}\mathbf{L}\boldsymbol{X} + c. \tag{17.156}$$

Therefore, we wish to find a preconditioner, $\mathbf{L}$, such that $\kappa\left(\mathbf{L}^T \mathbf{J}''\mathbf{L}\right) << \kappa\left(\mathbf{J}''\right)$. In fact it should be clear now what the preconditioner should be. We require $\mathbf{L}$ to be approximately equal to $\mathbf{J}''^{-\frac{1}{2}}$. If we

were able to have a preconditioner that was the square root of the Hessian, referred to as a **Hessian preconditioner**, then $\kappa\left(\mathbf{L}^T\mathbf{J}''\mathbf{L}\right)=1$, and as such the minimization converges in one iteration. This raises the question about whether or not there is an approximation to the Hessian preconditioner that can easily be inverted.

The Hessian matrix can be written in the form

$$\mathbf{J}''=\sum_{k=1}^{N}\lambda_k\boldsymbol{v}\boldsymbol{v}^T, \tag{17.157}$$

where λ_k and $\boldsymbol{v}_k$ are the eigenvalues and eigenvectors of the Hessian matrix, respectively. It is possible to define what is referred to as a **Hessian eigenvector preconditioner** as

$$\mathbf{L}^{-1}=\mathbf{I}+\sum_{k=1}^{K}\left(\mu_k^{\frac{1}{2}}-1\right)\boldsymbol{v}_k\boldsymbol{v}_k^T, \tag{17.158}$$

where we are only using K of the eigenvectors to build the preconditioner. We have to choose the parameter μ_k such that $\mu_k\lambda_k<\lambda_{K+1}$, which makes the condition number of the Hessian matrix of the preconditioned problem λ_{K+1}.

The eigenvectors can be computed by using the **Lanczos** method which is closely related to the conjugate gradient algorithm. We shall present the Lanczos methods in the next section.

17.4.1 Time-Parallel Preconditioning

In the last chapter we presented the saddle-point, or parallel in time, version of weak constraint 4D VAR. As we saw this was a significantly different formulation to the standard sequential cost function approach, but we did not address how to solve the system. This is addressed in [87], through a form of preconditioner. We summarize the work form [87] below.

We start by recalling the state formulation of weak constraint 4D VAR, where the analysis is the minimizer of the nonlinear cost function

$$\begin{aligned}J\left(\mathbf{x_0},\mathbf{x_1},\ldots,\mathbf{x_N}\right)=&\frac{1}{2}\left|\left|\mathbf{x_0}-\mathbf{x_b}\right|\right|^2_{\mathbf{B}^{-1}}+\frac{1}{2}\sum_{i=0}^{N}\left|\left|\mathbf{y_i}-\mathbf{h}\left(\mathbf{x_i}\right)\right|\right|^2_{\mathbf{R_i}^{-1}}\\&\frac{1}{2}\sum_{i=0}^{N-1}\left|\left|\mathbf{x_{i+1}}-\mathcal{M}_i\left(\mathbf{x_i}\right)\right|\right|_{\mathbf{Q_{i+1}}^2},\end{aligned} \tag{17.159}$$

where the norm notation above is defined as $||\mathbf{a}||^2_{\mathbf{A}}\equiv\mathbf{a}^T\mathbf{A}\mathbf{a}$.

In [87] it is now stated that the minimizer of (17.159) can be approximated using an inexact Gauss-Netwon, see next section for more details, algorithm. In an incremental approach, the $(j+1)\,th$ approximation $\underline{\mathbf{x}}^{(j+1)}=\left(\mathbf{x_0}^{(j+1)T},\mathbf{x_1}^{(j+1)T},\ldots,\mathbf{x_1}^{(N)T}\right)^T\in\mathbb{R}^{(N+1)n}$ of the state is

$$\underline{\mathbf{x}}^{(j+1)}=\underline{\mathbf{x}}^{(j)}+\delta\underline{\mathbf{x}}^{(j)} \tag{17.160}$$

where the update is $\delta\underline{\mathbf{x}}^{(j+1)}=\left(\delta\mathbf{x_0}^{(j+1)T},\delta\mathbf{x_1}^{(j+1)T},\ldots,\delta\mathbf{x_1}^{(N)T}\right)^T\in\mathbb{R}^{(N+1)n}$.

The next step is to introduce the four-dimensional arrays, as we saw for the saddle point derivation, but the matrix $\underline{\mathbf{F}}^{-1}$ is replaced by $\underline{\mathbf{L}}^{(j)}$ here but the rest of the notation is as before except for

$$\underline{\mathbf{b}}^{(j)} \equiv \begin{pmatrix} \mathbf{x_0}^{(j)} - \mathbf{x_b} \\ \mathcal{M}\left(\mathbf{x_0}^{(j)}\right) - \mathbf{x_1}^{(j)} \\ \vdots \\ \mathcal{M}\left(\mathbf{x_0}^{(j)}\right) - \mathbf{x_1}^{(j)} \end{pmatrix}, \tag{17.161}$$

$$\underline{\mathbf{d}}^{(j)} \equiv \begin{pmatrix} \mathbf{y_1} - \mathbf{h_1}\left(\mathbf{x_1^{(j)}}\right) \\ \vdots \\ \mathbf{y_N} - \mathbf{h_N}\left(\mathbf{x_N^{(j)}}\right) \end{pmatrix}. \tag{17.162}$$

This then leads to the update cost function as

$$J^{\delta}\left(\delta\underline{\mathbf{x}}^{(j)}\right) = \frac{1}{2}\left|\left|\underline{\mathbf{L}}^{(j)}\delta\underline{\mathbf{x}}^{(j)} - \underline{\mathbf{b}}^{(j)}\right|\right|^2_{\underline{\mathbf{D}}^{-1}} + \frac{1}{2}\left|\left|\underline{\mathbf{H}}^{(j)}\delta\underline{\mathbf{x}}^{(j)} - \underline{\mathbf{d}}^{(j)}\right|\right|^2_{\underline{\mathbf{R}}^{-1}}. \tag{17.163}$$

As a result of the cost function in (17.163) being quadratic, $\delta\underline{\mathbf{x}}^{(j)}$ can be found by solving the following large linear system with the Hessian matrix, $\underline{\mathbf{A}}^{(j)}$ of $J^{\delta}\left(\delta\underline{\mathbf{x}}^{(j)}\right)$:

$$\underline{\mathbf{A}}^{(j)}\delta\underline{\mathbf{x}}^{(j)} = \left(\underline{\mathbf{L}}\right)^{(j)}\underline{\mathbf{D}}^{-1}\underline{\mathbf{b}}^{(j)} + \left(\underline{\mathbf{H}}\right)^{(j)}\underline{\mathbf{R}}^{-1}\underline{\mathbf{d}}^{(j)}, \tag{17.164}$$

where

$$\underline{\mathbf{A}}^{(j)} \equiv \left(\underline{\mathbf{L}}^T\right)^{(j)}\underline{\mathbf{D}}^{-1}\underline{\mathbf{L}}^{(j)} + \left(\underline{\mathbf{H}}^T\right)^{(j)}\underline{\mathbf{R}}^{-1}\underline{\mathbf{H}}^{(j)}. \tag{17.165}$$

An assumption is now made in [87] that the total number of observations p is much less that all of the realizations, $(N+1)n$, thus $\left(\underline{\mathbf{H}}^T\right)^{(j)}\underline{\mathbf{R}}^{-1}\underline{\mathbf{H}}^{(j)}$ is symmetric positive semi-definite, while $\left(\underline{\mathbf{L}}^T\right)^{(j)}\underline{\mathbf{D}}^{-1}\underline{\mathbf{L}}^{(j)}$ is symmetric positive definite, this then make the Hessian matrix, which is dimensions $(N+1)n \times (N+1)n$, and is symmetric positive definite.

With regards to the preconditioner here the idea, as stated in [87], is to apply the preconditioner so that the first term of the preconditioned Hessian is equal to identity. Then the preconditioned Hessian is a sum of the identity matrix and a low-rank symmetric positive semi-definite matrix with rank at most p. Its smallest eigenvalue is equal to 1 and it has at most p eigenvalues that are larger than 1.

Applying this kind of preconditioning to the state formulation of weak constraint 4D VAR requires preconditioning with $\underline{\mathbf{L}}^{-1}\underline{\mathbf{D}}^{-\frac{1}{2}}$, where $\underline{\mathbf{L}}^{-1}$ the same as the definition for $\underline{\mathbf{F}}$ in the last chapter. It is then stated that Matrix-vector products with this 4D array are sequential in the time dimension.

Given this motivation, we are seeking an approximation $\widetilde{\underline{\mathbf{L}}}^{-1}$ to $\underline{\mathbf{L}}^{-1}$. As a result of this the precondition system to be solved is

$$\underline{\mathbf{A}}^{pr}\delta\widetilde{\underline{\mathbf{x}}} = \underline{\mathbf{D}}^{\frac{1}{2}}\widetilde{\underline{\mathbf{L}}}^{-T}\left(\underline{\mathbf{L}}^T\underline{\mathbf{D}}^{-1}\underline{\mathbf{b}} + \underline{\mathbf{H}}^T\underline{\mathbf{R}}^{-1}\underline{\mathbf{d}}\right), \tag{17.166}$$

where

$$\underline{\mathbf{A}}^{pr} \equiv \underline{\mathbf{D}}^{\frac{1}{2}} \widetilde{\underline{\mathbf{L}}}^{-T} \left(\mathbf{L}^T \underline{\mathbf{D}}^{-1} \underline{\mathbf{L}} + \underline{\mathbf{H}}^T \underline{\mathbf{R}}^{-1} \underline{\mathbf{H}} \right) \widetilde{\underline{\mathbf{L}}}^{-1} \underline{\mathbf{D}}^{\frac{1}{2}}, \tag{17.167}$$

along with

$$\widetilde{\underline{\mathbf{L}}}^{-1} \underline{\mathbf{D}}^{\frac{1}{2}} \delta \widetilde{\underline{\mathbf{x}}} = \delta \underline{\mathbf{x}}. \tag{17.168}$$

With the appropriate choice for $\widetilde{\underline{\mathbf{L}}}^{-1}$, $\underline{\mathbf{A}}^{pr}$ is symmetric positive definite, but should also be selected so that it can be ran in parallel. However, this not straightforward, and it shown in [124] that they were not able to find a sufficient choice.

In [87] they introduce a randomized precondition because they require matrix products with blocks of vectors that can be easily parallelized, and it is mentioned in [87] that it has been shown to be good approximation for matrices with rapidly decaying singular vectors.

A low rank approximation of $\underline{\mathbf{L}}^{-1}$ cannot be used as it would make the preconditioner $\underline{\mathbf{A}}^{pr}$ low rank and thus singular. [87] suggest exploiting the structure of $\underline{\mathbf{L}}^{-1}$, and to write it as

$$\underline{\mathbf{L}}^{-1} = \underline{\mathbf{I}} + \underline{\mathbf{P}}, \tag{17.169}$$

where $\underline{\mathbf{P}}$ is a strictly lower triangular matrix. It is then proposed to use a rank k approximation $\widetilde{\underline{\mathbf{P}}} = \underline{\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}}$ and produce a truncated singular value decomposition of size k. The bases of the work in [87] is to use randomised singular value decomposition to approximate the tangent linear model. This approach is a block method that is easy to parallelize in the sense that it requires calculating matrix products with blocks of vectors. Because [244] showed that it is important to take into account the information on the background errors when using model reduction techniques in data assimilation, the approach form [87] approximates the tangent linear model in interaction with the background- and model-error covariance matrices. The reader is referred to [87] for more of the algorithmic implementation of this approach and results with the Lorenz 96 model.

Given the motivation for why we wish to precondition, we now move on to consider different minimization, or optimization, algorithms as they are also referred to in some fields.

17.5 Minimization Algorithms

At the heart of variational-based data assimilation is a cost function that has to be minimized to find a solution. We know that this cost function could be nonlinear in terms of the analysis state that we seek, and/or very large, which implies that it is not possible to directly invert the matrices to find the solution, and that we have to iterate to find the solution. If we have managed to precondition the problem then the cost function should have a more circular structure to it. We now consider four minimization methods, increasing in complexity, to find the minimum state.

17.5.1 Newton-Raphson

To implement the Newton-Raphson algorithm, we require the Jacobian vector and the Hessian matrix, ∇J and $\nabla^2 J$ respectively. However, if we have a very large dimensional problem, storing the Hessian

matrix is not feasible, but also the Hessian matrix is a function of the iterate that we are considering, so it has to be updated at each iteration.

If we consider the univariate situation, and if we have the function $f(x)$ and we wish to find its root of $f(x) = 0$, $x = \alpha$, and we have a estimate of the root, x_0, then the Newton-Raphson method will generate a sequence of iterates, $x^{(n)}$, that we hope will converge to the root α. Since x_0 is assumed to be close to α, we approximate the function by constructing tangent lines at $(x_0, f(x_0))$. We then use the root of the tangent line to approximate α, where this new estimate is labeled $x^{(1)}$, and we carry on until some stopping criteria has been met.

The iteration formula for the Newton-Raphson is given by

$$x^{(n+1)} = x^{(n)} - \frac{f\left(x^{(n)}\right)}{f'\left(x^{(n)}\right)}. \tag{17.170}$$

However, in data assimilation applications it is not the root of the cost function that we seek, but that of its gradient. Therefore, we need a slight variation of (17.170). Thus, for our application the version of the Newton-Raphson method we require is

$$x^{(n+1)} = x^{(n)} - \frac{f'\left(x^{(n)}\right)}{f''\left(x^{(n)}\right)}. \tag{17.171}$$

However, we do not minimize a cost function with a single variable in data assimilation, so we require a higher dimensional version of (17.171), where if we denote $\boldsymbol{g}$ as the Jacobian of our function of vectors $f(\boldsymbol{x})$, and $\mathbf{H}$ as the Hessian matrix, then the multidimensional version of (17.171) is

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - \mathbf{H}^{-1}\left(\boldsymbol{x}^{(n)}\right)\boldsymbol{g}\left(\boldsymbol{x}^{(n)}\right). \tag{17.172}$$

However, as we saw in the last subsection, if we have introduced a change of variable such that the Hessian is the identity matrix then the Newton-Raphson method converges in one iteration.

It is highly unlikely that we would be able to store, or develop, a full version of the Hessian preconditioning but there have been research to find approximations to mimic Hessian preconditioning [506,511].

One approach to overcome the problem of finding the Hessian matrix, and its inverse, is to apply a finite difference type approach to the Hessian. This variation of the Newton-Raphson approach is referred to as the **secant method** and for the univariate is given by

$$x^{(n+1)} = x^{(n)} - f\left(x^{(n)}\right)\frac{x^{(n)} - x^{(n-1)}}{f\left(x^{(n)}\right) - f\left(x^{(n-1)}\right)}, \tag{17.173}$$

where we require the values of two points before we can start this approach. For our problem in the univariate case, (17.173) becomes

$$x^{(n+1)} = x^{(n)} - f'\left(x^{(n)}\right)\frac{x^{(n)} - x^{(n-1)}}{f'\left(x^{(n)}\right) - f'\left(x^{(n-1)}\right)}. \tag{17.174}$$

It is the approximation to the Hessian matrix that leads to the set of **Quasi-Newton** methods that have been designed over the years to deal with large dimensional problems.

17.5.2 Quasi-Newton Methods

It is not always possible to obtain the Hessian matrix, or build a full inverse Hessian preconditioner, and as such we wish to consider algorithms that are similar to the secant method that only require the gradient. Such a set of methods are the **quasi-Newton** methods just mentioned.

The quasi-Newton methods are effectively performing successive measurement of the gradient, and as such they build a quadratic model of the objective function that, if these models are sufficiently good approximations, then this allow us to achieve **superlinear** convergence.

The first of these quasi-Newton schemes is the **Broyden-Fletcher-Goldfarb-Shanno (BFGS)** method. The starting point for the BFGS method is to consider the search direction $\mathbf{p}_k$ at the kth iteration is given by the analogue of the Newton-Raphson method as

$$\mathbf{J}''_{(k)}\mathbf{p}_k = -\mathbf{g}\left(\boldsymbol{x}^{(k)}\right), \tag{17.175}$$

where $\mathbf{J}''$ is an approximation to the Hessian matrix that is updated iteratively at each stage. A line search in the direction of $\mathbf{p}_k$ is used to find the next point $\boldsymbol{x}^{(k+1)}$. Instead of requiring the full Hessian matrix at point $\boldsymbol{x}^{(k+1)}$ to be computed as $\mathbf{J}''_{(k+1)}$, the approximate Hessian at iteration k is updated by the addition of two matrices

$$\mathbf{J}''_{(k+1)} = \mathbf{J}''_{(k)} + \mathbf{U}_{(k)} + \mathbf{V}_{(k)}, \tag{17.176}$$

where both $\mathbf{U}_{(k)}$ and $\mathbf{V}_{(k)}$ are symmetric rank-one matrices, where the rank-one assumption is implying that the matrix is formed as an outer product of a vector with itself, that is to say, $\mathbf{D} = \boldsymbol{a}\boldsymbol{a}^T$.

The algorithmic description of the BFGS method is presented in Algorithm 17.1.

Algorithm 17.1 BFGS algorithm

1. Obtain a search direction $\mathbf{p}_k$ by solving the equation
$$\mathbf{J}''_{(k)}\mathbf{p}_k - \boldsymbol{q}\left(\boldsymbol{x}^{(k)}\right).$$
2. Perform a line search to find an acceptable step size α_k in the direction that was found in Step 1, and then update the iteration as $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k\mathbf{p}_k$.
3. Set $\mathbf{s}_k = \alpha_k\mathbf{p}_k$.
4. Next, form the vector $\mathbf{z}$ which will be equivalent to an approximation to the Hessian as $\mathbf{z} = \boldsymbol{g}\left(\boldsymbol{x}^{(k+1)}\right) - \boldsymbol{g}\left(\boldsymbol{x}^{(k)}\right)$.
5. Finally, update the approximation to the Hessian matrix as
$$\mathbf{J}''_{(k+1)} = \mathbf{J}''_{(k)} + \frac{\mathbf{z}_k\mathbf{z}_k^T}{\mathbf{z}_k^T\mathbf{s}_k} - \frac{\mathbf{J}''_{(k)}\mathbf{s}_k\mathbf{s}_k^T\mathbf{J}''_{(k)}}{\mathbf{s}_k^T\mathbf{J}''_{(k)}\mathbf{s}_k}. \tag{17.177}$$

It is quite plausible to start the BFGS scheme with the identity for the Hessian. The impact of this is that on the first iteration we have a gradient descent; however, as we start to build up better representations of the Hessian matrix, the algorithm becomes more refined.

We should note here that we require the inverse of (17.177) to find the search direction in the first step above. Therefore it can be shown that the inverse of the Hessian matrix approximation in (17.177)

is given by

$$\left(\mathbf{J}''_{(k+1)}\right)^{-1} = \left(\mathbf{I} - \frac{\mathbf{s}_k \mathbf{z}_k^T}{\mathbf{z}_k^T \mathbf{s}_k}\right)\left(\mathbf{J}''_{(k)}\right)^{-1}\left(\mathbf{I} - \frac{\mathbf{z}_k \mathbf{s}_k^T}{\mathbf{z}_k^T \mathbf{s}_k}\right) + \frac{\mathbf{s}_k \mathbf{s}_k^T}{\mathbf{z}_k^T \mathbf{s}_k}. \tag{17.178}$$

An alternative way of writing (17.178) is in a vectorial form as

$$\left(\mathbf{J}''_{(k+1)}\right)^{-1} = \left(\mathbf{J}''_{(k)}\right)^{-1} + \frac{\left(\mathbf{s}_k^T \mathbf{z}_k + \mathbf{z}_k^T \left(\mathbf{J}''_{(k)}\right)^{-1} \mathbf{z}_k\right)\left(\mathbf{s}_k \mathbf{s}_k^T\right)}{\left(\mathbf{s}_k^T \mathbf{z}_k\right)^2} - \frac{\left(\mathbf{J}''_{(k)}\right)^{-1} \mathbf{z}_k \mathbf{s}_k^T + \mathbf{s}_k \mathbf{z}_k^T \left(\mathbf{J}''_{(k)}\right)^{-1}}{\mathbf{s}_k^T \mathbf{z}_k}. \tag{17.179}$$

As the number of iterations increase, the estimate of the Hessian matrix from (17.179) requires the build-up of memory to store each iteration of this approximation. In [320] a new version of the BFGS is proposed, referred to as the **Limited Memory BFGS (L-BFGS)**. We shall not present the derivation from [320], but the reader is referred to this paper for the details of this scheme as it is in use operationally at some centers due to its ability to set a memory limit, yet still retain the important characteristics of the quasi-Newton methods.

17.5.3 Steepest Descent

Suppose that we would like to find the minimum of a function $f(x)$, and the gradient of f is denoted by $g_k = g(x_k) = \nabla f(x_k)$. We now need to compute a step along a given search direction, d_k;

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, \ldots, \tag{17.180}$$

where the step length, α_k is chosen so that

$$\alpha_k = \arg\min_{\alpha} f(\alpha_k + \alpha_k d_k), \tag{17.181}$$

and arg min refers to the argument of the minimum for the given function.

For the steepest descent method, the search direction is given by $d_k = -\nabla f(x_k)$. The steepest descent algorithm can be found in Algorithm 17.2.

Algorithm 17.2 Steepest Descent Algorithm

Given an initial x_0, $d_0 = -g_0$, and a convergence tolerance tol
for $k = 0$ to $maxiter$ **do**
  Set $\alpha_k = \arg\min \phi(\alpha) = f(x_k) - \alpha_k g_k$
  $x_{k+1} = x_k - \alpha_k g_k$
  Compute $g_{k+1} = \nabla f(x_k + 1)$
  if $\|g_{k+1}\|_2 \leq tol$ **then**
    converged
  end if
end for

The two main computational advantages of the steepest descent algorithm is the ease with which a computer algorithm can be implemented and the low storage requirements necessary, $O(n)$. The main

work requirement is the line search required to compute the step length, αk and the computation of the gradient.

17.5.4 Conjugate Gradient

One of the drawbacks of the steepest descent method is that we could find ourselves back in the same direction as earlier steps. An approach to avoid obtaining the same direction is to pick a set of **orthogonal** directions, $\mathbf{d}_0, \mathbf{d}_1, \ldots, \mathbf{d}_{n-1}$. Under this situation we would take exactly one step; that step will be just the correct length to line up evenly with x.

If we consider Fig. 17.6, we can see that the first step correctly hit the vertical coordinate, and the second step corrects the horizontal mismatch to arrive at the root of the function. The feature to notice about the two directions in Fig. 17.6 is that they are orthogonal to each other. Therefore, in general we have that at each step we choose a point

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \mathbf{d}_k. \tag{17.182}$$

For us to find the value of α_k we use that fact that $\mathbf{e}^{(k+1)}$ should be orthogonal to $\mathbf{d}_i$, so that we do not return to a direction we have searched before. The vector $\mathbf{e}$ is the **error** and is defined as $\boldsymbol{x}^{(k)} - \boldsymbol{x}^t$. Therefore, using these conditions, we have

$$\begin{aligned} \mathbf{d}_k^T \mathbf{e}_{k+1} &= 0, \\ \Rightarrow \mathbf{d}_k^T (\mathbf{e}_k + \alpha_k \mathbf{d}_k) &= 0, \\ \Rightarrow \alpha_k &= -\frac{\mathbf{d}_k^T \mathbf{e}_k}{\mathbf{d}_k^T \mathbf{d}_k}. \end{aligned}$$

However, we have a problem with the expression above as we do not know $\mathbf{e}_k$ so we cannot calculate α_k. To overcome this problem we introduce the following definition for **A-orthogonal**.

Definition 17.6. Given two vectors $\mathbf{d}_i$ and $\mathbf{d}_j$ and a symmetric positive definite matrix $\mathbf{A}$, then the two vectors are said to be **A-orthogonal**, or **conjugate**, if

$$\mathbf{d}_i^T \mathbf{A} \mathbf{d}_j = 0. \tag{17.183}$$

We recall here that we are solving to find the minimum of the $J(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \mathbf{A}\boldsymbol{x} - \mathbf{b}\boldsymbol{x} + \mathbf{c}$, and as such we can define the **residual**, $\mathbf{r}_k$, as $\mathbf{r}_k \equiv \mathbf{b} - \mathbf{A}\boldsymbol{x}^{(k)}$, which can be shown to be $\mathbf{r}_k = -\mathbf{A}\mathbf{e}_k$. Therefore, the

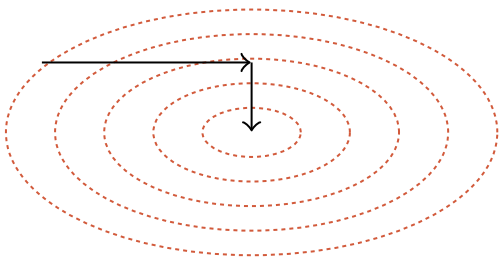


FIGURE 17.6

Schematic of the steps of the conjugate algorithm.

search direction is defined as

$$\alpha_k = \frac{\mathbf{d}_k^T \mathbf{r}_k}{\mathbf{d}_k^T \mathbf{A}\mathbf{d}_k}, \tag{17.184}$$

and we can see that is possible to calculate (17.184). There are many more tricks that are applied to arrive at the final algorithm for the conjugate gradient and we recommend [17] for a detailed description. The algorithm for the conjugate gradient method is shown in Algorithm 17.3.

Algorithm 17.3 Conjugate Gradient Algorithm

1. Calculate the initial residual and search directions as $\mathbf{r}_0 = \mathbf{A}\boldsymbol{x}^{(0)} - \mathbf{b}$, $\mathbf{d}_0 = -\mathbf{r}_0$ for an initial guess, $\boldsymbol{x}^{(0)}$.
2. Now iterate the following steps until the stopping criteria has been met:
3. $\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{d}_k^T \mathbf{A}\mathbf{d}_k}$,
4. $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \mathbf{d}_k$,
5. $\mathbf{r}_{k+1} = \mathbf{r}_k + \alpha_k \mathbf{A}\mathbf{d}_k$,
6. $\beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}$,
7. $\mathbf{d}_{k+1} = -\mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$.
8. Return to Step 3 unless the convergence criterion has been achieved.

A last remark about the conjugate gradient algorithm: while it is a fast method to find roots of equations, it is not always as fast as we would like. One way around this is to bring in a preconditioner. If a preconditioner for the $\mathbf{A}$ matrix is introduced, then the algorithm is now referred to as the **preconditioned conjugate gradient** algorithm.

17.5.5 Lanczos Methods

As mentioned earlier, the conjugate gradient and the Lanczos-based methods are quite similar. We shall only give a brief overview of the Lanczos approach for solving a least squares problem, but we do recommend reading the whole chapter in any of the versions of the Golub and Van Loan's books [157].

If we suppose that the matrix $\mathbf{A}$ is symmetric and positive definite, and we have the functional J, defined by

$$J(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \mathbf{A}\boldsymbol{x} - \boldsymbol{x}^T \mathbf{b},$$

and we know that the Jacobian of the cost function above is $\nabla J(\boldsymbol{x}) = \mathbf{A}\boldsymbol{x} - \mathbf{b}$, then the unique minimizer of J is $\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$. We now suppose that we have a initial guess $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$. We shall use the fact that one approach for obtaining a vector sequence, $\left\{\boldsymbol{x}^{(k)}\right\}$, that converges to the true minimum of the cost function is to generate a sequence of orthonormal vectors $\{\mathbf{q}_k\}$, which implies

$$\boldsymbol{x}^{(0)} + \text{span}\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k\} = \left\{\boldsymbol{x}^{(0)} + a_1\mathbf{q}_1 + a_2\mathbf{q}_2 + \ldots a_k\mathbf{q}_k\right\}, \tag{17.185}$$

where $a_k \in \mathbb{R}$, for $k = 1, 2, \ldots, N$. If we now form the matrix $\mathbf{Q}_k = [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_k]$, then this implies that for a vector $\boldsymbol{y} \in \mathbb{R}^k$, we have

$$J\left(\boldsymbol{x}^{(0)} + \mathbf{Q}_k \boldsymbol{y}\right) = \frac{1}{2}\left(\boldsymbol{x}^{(0)} + \mathbf{Q}_k \boldsymbol{y}\right)^T \mathbf{A}\left(\boldsymbol{x}^{(0)} + \mathbf{Q}_k \boldsymbol{y}\right) - \left(\boldsymbol{x}^{(0)} + \mathbf{Q}_k \boldsymbol{y}\right)^T \mathbf{b}. \tag{17.186}$$

Through differentiating (17.186) with respect to $\boldsymbol{y}$, and according to [157], we have

$$\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(0)} + \mathbf{Q}_k \boldsymbol{y}^{(k)}, \tag{17.187a}$$

$$\left(\mathbf{Q}_k^T \mathbf{A} \mathbf{Q}_k\right) \boldsymbol{y}^{(k)} = \mathbf{Q}_k^T \left(\mathbf{b} - \mathbf{A}\boldsymbol{x}^{(0)}\right). \tag{17.187b}$$

The Lanczos method is required to ensure that (1) the linear system in (17.187b) is easily solvable, and (2) we do not have to store all of the columns of $\mathbf{Q}_k$ to be able to update $\boldsymbol{x}^{(k)}$.

In [157] it is stated that after k steps of the Lanczos algorithm, we obtain the following factorization:

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_k \mathbf{T}_k + \mathbf{r}_k \mathbf{e}_k^T, \tag{17.188}$$

where $\mathbf{T}_k$ is a tridiagonal matrix which then makes (17.187b) is a symmetric positive definite tridiagonal system, which can be solved through a lower triangular factorization $\mathbf{L}\mathbf{D}\mathbf{L}^T$. Therefore, the basis of the Lanczos algorithm applied to our type of problem is finding values for these entries in the $\mathbf{L}$ and $\mathbf{D}$ matrices. The algorithm description can be found in [157] on page 493.

We now move on to the question of how to assess how well the data assimilation scheme has performed.

17.6 Performance Metrics

When designing and implementing a data assimilation scheme, it is imperative that we have ways to be able to determine the efficiency and accuracy of the scheme. In this section we present some different measures that are used to determine the required performance.

Root mean square error

One of the most commonly used methods to assess the accuracy of data assimilation schemes is the root means square error or RMSE. Note: There is also a ME which stands for Mean Error, which we shall define first and then follow with the RME.

If we assume that we have a true state, x^t, that could be a single value of it could be a vector, and we have our final analysis from a data assimilation scheme, and let x^a denote the analysis of the converged data assimilation scheme, then the RME and RMSE can be defined spatially, temporally, or both. The definitions for the RME and the RMSE are

$$RME = \frac{1}{N}\sum_{i=1}^{N}\left(x_i^t - x_i^a\right), \tag{17.189a}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(x_i^t - x_i^a\right)^2}, \tag{17.189b}$$

respectively. We should note that the quantity we are assessing against, i.e., the true state, may not be the analysis state but could be a forecast generated from an analysis state.

However, we should also note that the mean error could incorrectly be equal to zero, as a result of it being an average measure. This means that errors of opposite sign could cancel. This measure is a good indicator of whether or not we have a bias in the algorithm. Recall that we have assumed that the analysis errors are unbiased, implying that their mean error should be zero.

It is possible to build up climatologies of these measures, time series to determine a trend, or use them to determine a systematic bias associated with some component of the data assimilation scheme. The aim of the data assimilation scheme is to be converging to the true state as we receive more information, and as such we would be looking to verify that the data assimilation scheme is converging to the true state. Therefore, when plotting RME or RMSE as a time series, we would hope to see a decaying curve toward zero. A constant difference in the time series, where there appears to be no decaying error structure, is also an indicator of a bias, while finally an increasing time series of the two types of error measurement here indicated that the assimilation scheme is **diverging**, or is **unstable**.

Another continuous measure that is often used to assess the performance of the data assimilation scheme is the **bias**, where mathematically this is defined by

$$bias = \frac{\frac{1}{N}\sum_{i=1}^{N} x_i^a}{\frac{1}{N}\sum_{i=1}^{N} x_i^t}. \tag{17.190}$$

However, it is often the case in the real world that we do not have the true state, and as such the three measures are usually assessed against observations of the situation; often it is a forecast that has started from the analysis state that we are more interested in. If we are using the analysis or forecast state, then these measures are labeled with analysis or forecast in front of them respectively.

However, it is quite often the case that we are assessing if specific features were predicted correctly, but these are now **dichotomous** or **categorical**. The main display for these variables is through a **contingency table**. The contingency table for the two-by-two case is presented in Table 17.1.

Given the number of hits, misses, false alarms, and correct negatives, we can define N, the total number of cases, as $N = H + M + FA + CN$. The first statistic that we present is the **accuracy**, which is defined as

$$Accuracy = \frac{H + CN}{N}. \tag{17.191}$$

The interpretation of this statistic is to quantify what fraction of the forecasts were correct.

The next two statistics are advised to be analyzed together; these scores are the **probability of detection** or **POD**, and the **false alarm rate**, or **FAR**, and are defined as

Table 17.1 Two-by-Two Contingency Table.

Observed Forecasted	Yes	No
Yes	Hits (H)	False alarms (FA)
No	Misses (M)	Correct negatives (CN)

$$POD = \frac{H}{H+M}, \tag{17.192a}$$

$$FAR = \frac{FA}{H+FA}, \tag{17.192b}$$

where the POD measures the fraction of correctly predicted yes events, while the FAR measures the fraction of forecasted yes events but were actually nos.

There are some more advanced statistics that can be calculated to ascertain how well events have or have not been forecasted. The three that we present below are the **equitable threat score**, or **ETS**, which is sometimes called the **Gilbert skill score**, the **Heidke skill score** or **HSSs**, and finally the **Odds ratio skill score** or **ORSS**.

$$ETS = \frac{H - H_{random}}{H+M+FA-H_{random}}, \tag{17.193a}$$

$$HSS = \frac{N + CN - EC_{random}}{N - EC_{random}}, \tag{17.193b}$$

$$ORSS = \frac{H \times CN - M \times FA}{H \times CN + M \times FA}, \tag{17.193c}$$

where H_{random} is given by

$$H_{random} = \frac{(H+M)(H+FA)}{H+M+FA+CN},$$

and EC_{random}, which stands for *expected correct* is given by

$$EC_{random} = \frac{1}{N}\left((H+M)(H+FA) + (CN+M)(CN+FA)\right).$$

The three statistics measure the following: (1) the ETS measures how well the forecasted predicted the yes event, while accounting for the hits due to chance; (2) the HSS measures the accuracy of the forecast, relative to that of random chance; and finally (3) the ORSS statistic measures the improvement of the forecast over random chance.

While we have only considered the two category situation here, it is possible that we could have multiple values of ranges of a variable that we are forecasting and observing, and there are also measures that can be applied to this situation.

17.6.1 Scorecard

There has been a lot of development at the different operational numerical weather prediction centers to provide a visual aid to indicate the changes in the performance in the data assimilation system on certain variables. Each center has a different way of presenting this, but we have a copy of a score card which is Figure 10 in [56] in Fig. 17.7 that shows the difference between two changes to aspects of the data assimilation system and how they impacted the forecast and different lead times, where here blue means bad, red means good and the larger the triangle the worse or better the impact is.

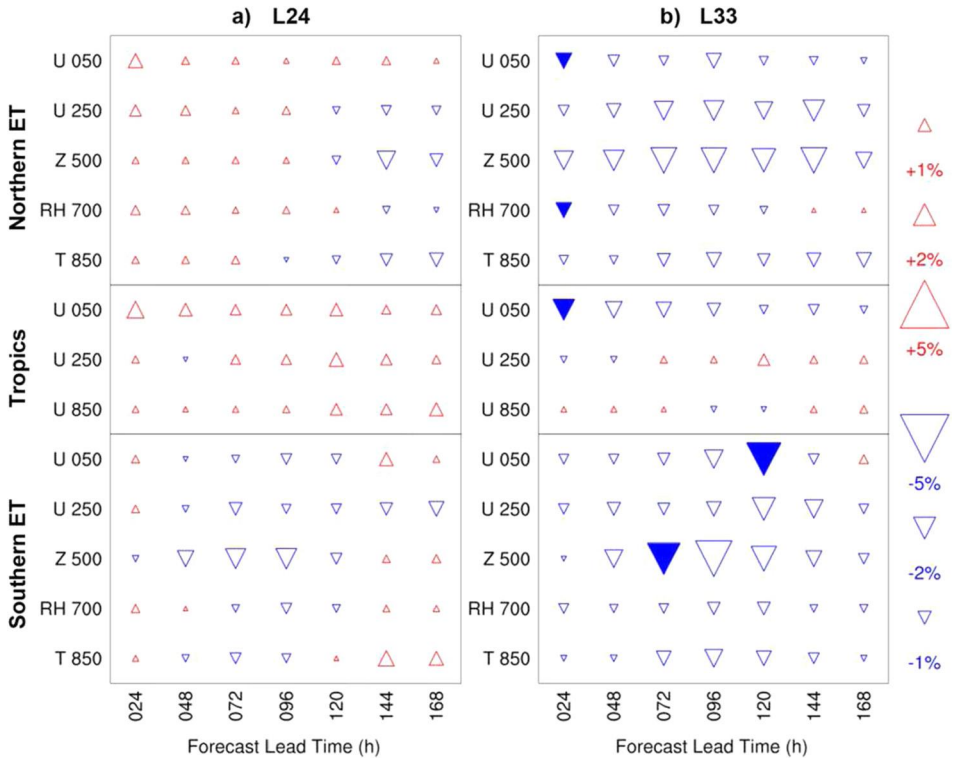


FIGURE 17.7

Copy of figure 10 showing a score card for some experiment from Caron, J., & Buehner, M. (2018). Scale-Dependent Background Error Covariance Localization: Evaluation in a Global Deterministic Weather Forecasting System, Monthly Weather Review, 146(5), 1367-1381. **©Used with permission.**

17.7 Summary

In this chapter we have introduced some of the subcomponents involved in implementing a variational-based data assimilation scheme. We have highlighted the need to ensure that the analysis state that will be used to initialize the next forecast is filtered so that the initial conditions will not excite spurious inertia gravity modes, which is what thwarted Richardson on his first attempt to forecast the weather. We have mentioned three different techniques to try to meet the objective of preventing the excitement of these spurious gravity modes: linear and nonlinear normal mode initialization, and digital filtering and balance constraints, where these constraints could either be the control variable transform or through the background error covariance model.

We have presented some covariance models that have been, or are being, used in operational numerical weather and ocean prediction. We have presented a couple of methods that have been used to estimate the background error variances and covariances, as well as preconditioning techniques to aid

in the minimization of the cost function. We have also introduced selection of minimization algorithms that can be used to find the minimum state of the cost function, along with performance metrics, and the operational scorecard.

We now move on to consider a different form of variational data assimilation, but instead of minimizing the cost function in model space, we minimize the cost function in observation space. These approaches are referred to as the **Physical Space Assimilation Systems**, or PSAS.

CHAPTER

Observation Space Variational Data Assimilation Methods

18

Contents

The physical-space component of the name of these types of variational data assimilation comes from the formulation of the background error covariance matrix being in physical (grid-point) model space, rather than in spectral space [328]. The name physical-space statistical analysis system (PSAS) comes from the 1998 paper by Dr. Stephen Cohn and coauthors from the Global Assimilation and Modeling Office at the NASA Goddard Space Flight Center [69]. The motivation to move away from the optimum interpolation (OI) methods was to avoid restrictions on the volumes of observations that could be used to form the analysis state [69,258]. The restrictions of the observations was through data selection, and the restriction that only observations at each grid point, or in each volume, within a neighborhood of the grid point, or volume, could be used.

Another difference between the PSAS-based data assimilation system and the spectral-based data assimilation system, is that they are optimized in observation space. Note that since the time of the original writing of the 1998 paper, quite a few operational data assimilation systems no longer form their covariance models in spectral space, but do run their nonlinear model in spectral space.

As well as PSAS being developed at NASA in the late 1990s, a similar version was being developed at the Naval Research Laboratory which was called the Naval Research Laboratory Atmospheric Variational Data Assimilation System, or NAVDAS [86].

18.1 Derivation of Observation Space-Based 3D VAR

We can either arrive at the PSAS equations through the minimum variance approach, which is the approach that we followed for optimal interpolation, or we can follow the maximum likelihood approach as we did for 3D VAR. We shall summarize the 3D VAR-based approach from [86] here.

At some time t, we define $\boldsymbol{x}^t$, $\boldsymbol{x}_b$, and $\boldsymbol{x}_a$ as the truth, background, and analysis states, respectively, where these vectors are defined on a regular grid $\mathbf{r}_i$, $i = 1, 2, \ldots, N$. As with the 3D VAR derivation we defined the background error as $\boldsymbol{\varepsilon}_b = \boldsymbol{x}^t - \boldsymbol{x}_b$, and we assume that the background error is unbiased,

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00022-X

which implies that $\mathbb{E}[\boldsymbol{\varepsilon}_b] = \mathbf{0}$. Therefore, the symmetric positive definite background error covariance matrix, $\mathbf{P}_b$, is defined as

$$\mathbf{P}_b \equiv \mathbb{E}[\boldsymbol{\varepsilon}_b \boldsymbol{\varepsilon}_b^T]. \tag{18.1}$$

We now introduce a column vector of observations, $\boldsymbol{y}$, of dimension N_o, where the observations may be direct or indirect measurements of the background variables. Next we introduce the observation operator, $\boldsymbol{h}(\boldsymbol{x}^t)$, such that

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}^t) + \boldsymbol{\varepsilon}_o, \tag{18.2}$$

where $\boldsymbol{\varepsilon}_o$ is the observational error. It is assumed that the observational error is unbiased, and that the observation error covariance matrix, $\mathbf{R}$, is defined as

$$\mathbf{R} \equiv \mathbb{E}[\boldsymbol{\varepsilon}_o \boldsymbol{\varepsilon}_o^T]. \tag{18.3}$$

Next we assume that the background and observational errors are uncorrelated, and that they both follow Gaussian distributions with mean zero but different variances. Therefore, given the Bayesian formulation from 3D VAR, we know that the maximum likelihood state is the minimum of

$$J(\boldsymbol{x}_a) = \frac{1}{2}(\boldsymbol{x}_a - \boldsymbol{x}_b)^T \mathbf{P}_b (\boldsymbol{x}^t - \boldsymbol{x}_b) + \frac{1}{2}(\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_a))^T \mathbf{R}^{-1} (\boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_a)). \tag{18.4}$$

We now linearize the observation operator about the background state, and apply a Taylor series expansion to the first order as

$$\boldsymbol{h}(\boldsymbol{x}_a) \approx \boldsymbol{h}(\boldsymbol{x}_b) + \mathbf{H}(\boldsymbol{x}_a - \boldsymbol{x}_b), \tag{18.5}$$

where $\mathbf{H}$ is the tangent linear model of the observation operator and is defined as

$$\mathbf{H} \equiv \left. \frac{\partial \boldsymbol{h}(\boldsymbol{x}_b)}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x}=\boldsymbol{x}_b}, \tag{18.6}$$

which implies that we can rewrite (18.4) as

$$J(\boldsymbol{x}_a) = \frac{1}{2}(\boldsymbol{x}_a - \boldsymbol{x}_b)^T (\mathbf{P}_b + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})(\boldsymbol{x}_a - \boldsymbol{x}_b) - (\boldsymbol{x}_a - \boldsymbol{x}_b)^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{d} + \frac{1}{2} \mathbf{d} \mathbf{R}^{-1} \mathbf{d}, \tag{18.7}$$

where $\mathbf{d} = \boldsymbol{y} - \boldsymbol{h}(\boldsymbol{x}_b)$.

In [86] the term $\mathbf{d}$ is the **observation increment**, and $\boldsymbol{x}_a - \boldsymbol{x}_b$ is the **analysis increment**.

If we now differentiate (18.7) with respect to $\boldsymbol{x}_a$ to find the minimum of the cost function, then the Jacobian of (18.7) is given by

$$\nabla_{\boldsymbol{x}_a} J(\boldsymbol{x}_a) = (\mathbf{P}_b + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})(\boldsymbol{x}_a - \boldsymbol{x}_b) - \mathbf{H}^T \mathbf{R}^{-1} \mathbf{d}. \tag{18.8}$$

To find the minimum of (18.7) we require the value of $\boldsymbol{x}_a$ so (18.8) is equal to $\mathbf{0}$. Therefore, setting (18.8) to zero and rearranging yields a global OI type solution given by

$$\boldsymbol{x}_a - \boldsymbol{x}_b = (\mathbf{P}_b + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{d}. \tag{18.9}$$

However, the expression in (18.9) is still in term of model space dimensions which involves the inversion of the very large matrix $(\mathbf{P}_b + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})$. As indicated in Chapter 16, it is possible to recast (18.9) into observation space through the **Sherman-Morrison-Woodbury** formula.

Recalling the formula, we have

$$(\mathbf{A}+\mathbf{UC}^{-1}\mathbf{V})^{-1} \equiv \mathbf{A}^{-1} + \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1}. \tag{18.10}$$

While your instincts might lead you to let $\mathbf{A} = \mathbf{P}_b$, $\mathbf{U} = \mathbf{H}^{\mathbf{T}}$, $\mathbf{C} = \mathbf{R}^{-\mathbf{1}}$, and $\mathbf{V} = \mathbf{H}$, we have to remember that the matrix we wish to invert in (18.9) is multiplied on the right by $\mathbf{H}^T\mathbf{R}^{-1}$ and we need to take that into account. Thus we shall take $\mathbf{A} = \mathbf{P}_b$ but we shall take $\mathbf{U} = \mathbf{H}^T\mathbf{R}^{-1}$, $\mathbf{C} = \mathbf{I}$, and $\mathbf{V} = \mathbf{H}$. Therefore, substituting these choices into (18.10) and then into (18.9) results in

$$(\mathbf{P}_b^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1} = (\mathbf{P}_b - \mathbf{P}_b\mathbf{H}^T\mathbf{R}^{-\mathbf{1}}(\mathbf{I}+\mathbf{HP}_b\mathbf{H}^T)^{-1}\mathbf{HP}_b)\mathbf{H}^T\mathbf{R}^{-1}. \tag{18.11}$$

Now if we multiply on the right by $\mathbf{H}^T\mathbf{R}^{-1}$ and then factorize $\mathbf{P}_b\mathbf{H}^T$ on the left, we have

$$(\mathbf{P}_b^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{P}_b\mathbf{H}^T(\mathbf{R}^{-1} - \mathbf{R}^{-1}(\mathbf{I}+\mathbf{HP}_b\mathbf{H}^T)^{-1}\mathbf{HP}_b\mathbf{H}^T\mathbf{R}^{-1}). \tag{18.12}$$

We now apply the Sherman-Morrison-Woodbury formula again, but this time in reverse and set $\mathbf{A} = \mathbf{R}^{-1}$, $\mathbf{U} = \mathbf{I}$, $\mathbf{V} = \mathbf{HP}_b\mathbf{H}^T$, and $\mathbf{C} = \mathbf{I}$. This implies that we can write (18.12) as

$$(\mathbf{P}_b^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1} = \mathbf{P}_b\mathbf{H}^T(\mathbf{R}+\mathbf{HP}_b\mathbf{H}^T)^{-1}, \tag{18.13}$$

and hence we are now in observation space and not model space. While we still do not perform the actual inversion in (18.13), we can now write a set of linear equations in observation space for the analysis in two steps as highlighted in [69,86]. The first step is to find an intermediate vector z such that

$$(\mathbf{HP}_b\mathbf{H}^T)z = \mathbf{d}, \tag{18.14}$$

and then perform a **post-multiplication** step by left-multiplying z by $\mathbf{P}_b\mathbf{H}^T$ such that

$$\boldsymbol{x}_a - \boldsymbol{x}_b = \mathbf{P}_b\mathbf{H}^T z. \tag{18.15}$$

The important feature of (18.14) is that we are attempting to solve a set of N_o linear equations, whereas in 3D VAR we are solving N linear equations, where there is usually a couple of orders of magnitude difference between N_o and N. The matrix on the left-hand side of (18.14) is the **innovation covariance matrix** and is a symmetric positive definite matrix.

It is stated in both [69] and [86] that the innovation covariance matrix, denoted by $\mathbf{M}$, [69], is not sparse due to the different covariance/correlation models that have been used. To introduce some sparseness into $\mathbf{M}$, both [69] and [86] divide the globe up into different regions.

In [69] the globe is partitioned into 80 equal-area regions using an icosahedral grid, while in [86] it is the observations that are divided up into prisms that contain approximately the same number of observations. This means that these prisms are not equal in area but are proportional in size to the density of the observation network in specific regions. As we have alluded to above, the reason for these partitions is to apply different forms of preconditioning to $\mathbf{M}$. We shall not go into any more of the implementation features here, but we do recommend reading [69] and [85], as they provide very good details of the implementation of their approaches. At the end they are both solving the same set of equations; it is how they go about it that is different.

However, the derivation in this section has only been for the spatial dimension. We now move on to explain the theory that allowed NAVDAS to become NAVDAS-AR and then NAVGEM, which is the US Navy's 4D VAR system in observation space.

18.2 4D VAR in Observation Space

The basis of representer theory can be found in [29] and, depending on the application, the derivation to the set of coupled Euler-Lagrange equations that you need to decouple could be continuous or discrete [487]. We shall follow the discrete version, which is consistent with the derivation of the other data assimilation schemes presented so far.

As we saw with the 3D VAR transform into observation space, we started with the error definitions, their covariance matrices, and then seek the maximum likelihood state of the Bayesian problem as set out in [259]. We have not presented this yet, but it is possible to write 4D VAR in terms of a Bayesian problem [129]; however, we shall present this model in Chapter 21 as part of the extension to non-Gaussian error distribution.

Therefore, for 4D VAR observation based data assimilation, we start with the error definitions; we keep the background error definition from before, but now this is valid at an analysis time t_a, which may or may not be at the beginning of the window, or there could be several analysis times throughout a window. The observational errors are now valid at different times throughout the assimilation window, but the subtle change introduced in [369,487–489] is to the model error definition, which is defined at each time step. Therefore, the generalized inverse problem comprises of three components: the background error, J^b, the observational error, J^o, and the model error, J^m, such that the general cost function is written as

$$J(\boldsymbol{x}^a) = J_0^b + J^o + J^m, \tag{18.16a}$$

$$J_0^b = \frac{1}{2}(\boldsymbol{x}_0^b - \boldsymbol{x}_o)^T [\mathbf{P}_0^b]^{-1}(\boldsymbol{x}_0^b - \boldsymbol{x}_o), \tag{18.16b}$$

$$J^q = \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}(\boldsymbol{x}_n - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1}))^T \mathbf{Q}_{nn'}^{-1}(\boldsymbol{x}_{n'} - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n'-1})), \tag{18.16c}$$

$$J^r = \frac{1}{2}\sum_{n=0}^{N}\sum_{n'=0}^{N}(\boldsymbol{y}_n - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_n))^T \mathbf{R}_{nn'}^{-1}(\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'})), \tag{18.16d}$$

where $\mathbf{Q}_{nn'}$ is the model error block covariance matrix between time t_n and $t_{n'}$, $\boldsymbol{\mathcal{M}}(\boldsymbol{x}_n)$ is the discrete version of the nonlinear forecast model. We are also assuming at the moment that observations are correlated in time. $\mathbf{P}_0^b$ is the background error covariance matrix at the initial time; also $\mathbf{R}_{nn'}$ is the observational error covariance matrix between $t = t_n$ and $t = t_{n'}$.

We now apply calculus of variation theory and so take the first variation of (18.16b)–(18.16d), which results in

$$\begin{aligned}\delta J = &-[\delta\boldsymbol{x}_0]^T[\mathbf{P}_0^b]^{-1}(\boldsymbol{x}_0^b - \boldsymbol{x}_o)\\ &+\sum_{n=1}^{N}\sum_{n'=1}^{N}[\delta\boldsymbol{x}_n - \mathbf{M}_{n-1}\delta\boldsymbol{x}_{n-1}]^T\mathbf{Q}_{nn'}^{-1}(\boldsymbol{x}_n - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1})) - \sum_{n=0}^{N}\sum_{n'=0}^{N}[\mathbf{H}_n\delta\boldsymbol{x}_n]^T\mathbf{R}_{nn'}^{-1}(\boldsymbol{y}_n - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_n)),\end{aligned} \tag{18.17}$$

where $\mathbf{M}$ here is the Jacobian matrix of the nonlinear numerical model and $\mathbf{H}$ is the Jacobian matrix of the nonlinear observation operator.

The next step is to introduce an adjoint field, denoted by $\boldsymbol{\lambda}^T = (\lambda_1 \dots \lambda_n \dots \lambda_N)$, and is defined by

$$\boldsymbol{\lambda}_n \equiv \sum_{n'=1}^{N} \mathbf{Q}_{nn'}^{-1}(\boldsymbol{x}_n - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1})), \quad \text{for } 1 \le n \le N. \tag{18.18}$$

An observation-space vector is also introduced, $\mathbf{h} = (\mathbf{h}_0^T \dots \mathbf{h}_n^T \dots \mathbf{h}_N^T)$, defined by

$$\mathbf{h}_n \equiv \sum_{n'=0}^{N} \mathbf{R}_{nn'}^{-1}(\boldsymbol{y}_n - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_n)). \tag{18.19}$$

Substituting (18.18) and (18.19) into (18.17) results in

$$\delta J = -\delta \boldsymbol{x}_0^T [\mathbf{P}_0^b]^{-1}(\boldsymbol{x}_0^b - \boldsymbol{x}_o) + \sum_{n=1}^{N} [\delta \boldsymbol{x}_n - \mathbf{M}_{n-1}\delta \boldsymbol{x}_{n-1}]^T \boldsymbol{\lambda}_n - \sum_{n=0}^{N} [\mathbf{H}_n \delta \boldsymbol{x}_n]^T \mathbf{h}_n. \tag{18.20}$$

The next step is to collect like terms of the variations, but also to apply integration by parts to alter the index on one of the summations so that the summations are over the same indices, which yields

$$\delta J = -\delta \boldsymbol{x}_0^T [\mathbf{P}_0^b]^{-1}(\boldsymbol{x}_0^b - \boldsymbol{x}_o) + \sum_{n=1}^{N} \delta \boldsymbol{x}_n^T - \sum_{n=0}^{N-1} \delta \boldsymbol{x}_n^T \mathbf{M}_n^T \boldsymbol{\lambda}_{n+1} - \sum_{n=1}^{N} \delta \boldsymbol{x}_n^T \mathbf{H}_n^T \mathbf{h}_n - \delta \boldsymbol{x}_0^T \mathbf{H}_0^T \mathbf{h}_0. \tag{18.21}$$

Now applying the theory of calculus of variation implies that all of the three terms above have to be simultaneously equal to zero, which mean that all three terms multiplying a variation must be equal to zero at the analysis state. Therefore, we have the conditions

$$\boldsymbol{\lambda}_n^a - \mathbf{M}_n^T \boldsymbol{\lambda}_{n+1}^a = \mathbf{H}_n^T \mathbf{h}_n^a, \quad \text{for } 1 \le n \le N-1, \tag{18.22}$$

$$\boldsymbol{\lambda}_N^a = \mathbf{H}_N^T \mathbf{h}_N^a, \tag{18.23}$$

$$[\boldsymbol{x}_0^a - \boldsymbol{x}_0^b] = \mathbf{P}_0^b \{\mathbf{M}_0^T \boldsymbol{\lambda}_1^a + \mathbf{H}_0^T \mathbf{h}_0^a\}. \tag{18.24}$$

Expanding (18.22)–(18.24) with (18.18) and (18.19) leads to the nonlinear Euler-Lagrange system:

$$\boldsymbol{\lambda}_n^a - \mathbf{M}_n^T \boldsymbol{\lambda}_{n+1}^a = \mathbf{H}_n^T \sum_{n'=0}^{N} \mathbf{R}_{nn'}^{-1}[\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'}^a)], \quad \text{for } 1 \le n \le N-1,$$

$$\text{subject to} \quad \boldsymbol{\lambda}_N^a = \mathbf{H}_N^T \sum_{n'=0}^{N} \mathbf{R}_{Nn'}^{-1}[\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'}^a)], \tag{18.25}$$

and

$$\boldsymbol{x}_n^a - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1}^a) \equiv \sum_{n'=1}^{N} \mathbf{Q}_{nn'} \boldsymbol{\lambda}_{n'}^a, \quad \text{for } 1 \le n \le N,$$

$$\text{subject to} \quad [\boldsymbol{x}_0^a - \boldsymbol{x}_0^b] = \mathbf{P}_0^b \left\{ \mathbf{M}_0^T \boldsymbol{\lambda}_1^a + \mathbf{H}_0^T \sum_{n'=1}^{N} \mathbf{R}_{0n'}^{-1} \left[\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}} \left(\boldsymbol{x}_{n'}^a \right) \right] \right\}. \tag{18.26}$$

18.2.1 Solution to the Coupled Linear Euler-Lagrange System

The next part of the derivation to the 4D VAR formulation in observation space is to consider the linear system. This is where it is assumed that the forecast model and the observation operators are linear. This allows the model to be expressed as $\mathcal{M}(\boldsymbol{x}_n) = \mathbf{M}_n \boldsymbol{x}_n$ and the observation operator as $\mathcal{H}(\boldsymbol{x}_n) = \mathbf{H}_n \boldsymbol{x}_n$. It is now assumed that the model and observation errors are uncorrelated in time, which makes $\mathbf{Q}_{nn'} = \mathbf{0}_{II}$ and $\mathbf{R}_{nn'} = \mathbf{0}_{k_n k_{n'}}$ for $n \neq n'$, where the subscript I represents the total number of state variables and k_n is the total number of observations at time t_n. The final assumption made here is that the observations at t_0 are not to be assimilated in this window, as they have already been assimilated in the previous window and **therefore, should not be assimilated twice**. As a result of these assumptions (18.18), (18.25), and (18.26) can be rewritten as

$$\boldsymbol{\lambda}_n^a \equiv \mathbf{Q}_n^{-1}[\boldsymbol{x}_n^a - \mathbf{M}_{n-1}\boldsymbol{x}_{n-1}^a], \quad \text{for } 1 \leq n \leq N, \tag{18.27}$$

$$\boldsymbol{\lambda}_n^a - \mathbf{M}_n^T \boldsymbol{\lambda}_{n+1}^a = \mathbf{H}_n^T \mathbf{R}_n^{-1}[\boldsymbol{y}_n - \mathbf{H}(\boldsymbol{x}_n^a)], \quad \text{for } 1 \leq n \leq N-1, \tag{18.28}$$

$$\text{subject to} \quad \boldsymbol{\lambda}_N^a = \mathbf{H}_N^T \mathbf{R}_N^{-1}[\boldsymbol{y}_N - \mathbf{H}(\boldsymbol{x}_N^a)],$$

$$\boldsymbol{x}_n^a - \mathbf{M}_{n-1}\boldsymbol{x}_{n-1}^a \equiv \mathbf{Q}_n \boldsymbol{\lambda}_n^a, \quad \text{for } 1 \leq n \leq N, \tag{18.29}$$

$$\text{subject to} \quad [\boldsymbol{x}_0^a - \boldsymbol{x}_0^b] = \mathbf{P}_0^b \mathbf{M}_0^T \boldsymbol{\lambda}_1^a.$$

The approach used to solve (18.28) and (18.29) involves introducing a representer field, denoted by $\boldsymbol{\gamma}_k$, and its adjoint, $\boldsymbol{\alpha}_k$, for the kth observation. We also require $\boldsymbol{\gamma}_k$ and $\boldsymbol{\alpha}_k$ to satisfy the following equations:

$$(\boldsymbol{\alpha}_k)_n - [\mathbf{M}_n]^T (\boldsymbol{\alpha}_k)_{n+1} = [(\mathbf{H}_k)_n]^T, \quad \text{for } 1 \leq n \leq N-1, \tag{18.30}$$

$$\text{subject to} \quad (\boldsymbol{\alpha}_N)_n = [(\mathbf{H}_k)_N]^T,$$

$$(\boldsymbol{\gamma}_k)_n - [\mathbf{M}_{n-1}]^T (\boldsymbol{\gamma}_k)_{n-1} = [(\mathbf{Q}_k)_n (\boldsymbol{\alpha}_k)_n], \quad \text{for } 1 \leq n \leq N, \tag{18.31}$$

$$\text{subject to} \quad (\boldsymbol{\gamma}_N)_0 = \mathbf{P}_0^b [(\mathbf{M}_0)_N]^T (\boldsymbol{\alpha}_k)_1.$$

Now a series of arrays are introduced to simplify the calculations to all times. These are denoted as

$$[\boldsymbol{\alpha}_k]^T = [[(\boldsymbol{\alpha}_k)_1]^T \ldots [(\boldsymbol{\alpha}_k)_n]^T \ldots [(\boldsymbol{\alpha}_k)_N]^T],$$
$$[\boldsymbol{\gamma}_k]^T = [[(\boldsymbol{\gamma}_k)_1]^T \ldots [(\boldsymbol{\gamma}_k)_n]^T \ldots [(\boldsymbol{\gamma}_k)_N]^T],$$
$$[\mathbf{H}_k]^T = [[(\mathbf{H}_k)_1]^T \ldots [(\mathbf{H}_k)_n]^T \ldots [(\mathbf{H}_k)_N]^T].$$

There are four final arrays that are introduced to make the derivation compact so that it can be written as a series of matrix-vector multiplications. These arrays are

$$\mathbf{Q} = \begin{pmatrix} \mathbf{Q}_1 & \cdot & \cdot & \cdot & \mathbf{0} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \mathbf{Q}_n & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{0} & \cdot & \cdot & \cdot & \mathbf{Q}_N \end{pmatrix}, \tag{18.32}$$

$$\mathbf{Q}^* = \begin{pmatrix} \mathbf{M}_0\mathbf{P}_0^b\mathbf{M}_0^T + \mathbf{Q}_1 & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_2 & \mathbf{0} & \cdot & \cdot & \cdot & \cdot \\ \cdot & & & \cdot & & & \\ \cdot & & & \mathbf{Q}_n & \mathbf{0} & \cdot & \cdot \\ \cdot & & & \cdot & & & \\ \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \mathbf{Q}_{N-1} & \mathbf{0} \\ \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \mathbf{Q}_N \end{pmatrix}, \tag{18.33}$$

$$\mathbf{M}^T = \begin{pmatrix} \mathbf{I} & \prod_{n'=1}^{1} \mathbf{M}_{n'}^T & \cdot & \cdot & \cdot & \prod_{n'=1}^{N-1} \mathbf{M}_{n'}^T \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \mathbf{I} & \prod_{n'=n-1}^{n-1} \mathbf{M}_{n'}^T & \cdot & \prod_{n'=n-1}^{N-1} \mathbf{M}_{n'}^T \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \mathbf{I} \end{pmatrix}, \tag{18.34}$$

$$\mathbf{M} = \begin{pmatrix} \mathbf{I} & \cdot & \cdot & \cdot & \cdot & \mathbf{0} \\ \prod_{n'=1}^{1} \mathbf{M}_{n'} & \mathbf{I} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \prod_{n'=n-1}^{1} \mathbf{M}_{n'} & \cdot & \prod_{n'=n-1}^{n-1} \mathbf{M}_{n'} & \mathbf{I} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \prod_{n'=N-1}^{1} \mathbf{M}_{n'} & \cdot & \cdot & \cdot & \prod_{n'=N-1}^{N-1} \mathbf{M}_{n'} & \mathbf{I} \end{pmatrix}. \tag{18.35}$$

The definition of the arrays above enable (18.30) and (18.31) to be written as

$$\boldsymbol{\alpha}_k = \mathbf{M}^T\mathbf{H}_k^T, \tag{18.36}$$

$$\boldsymbol{\gamma}_k = \mathbf{M}\mathbf{Q}^*\boldsymbol{\alpha}_k. \tag{18.37}$$

A final array that we introduce to help with capturing of all of the arrays involved is denoted as $\mathbf{P}^b = \mathbf{M}\mathbf{Q}^*\mathbf{M}^T$. This new array enables representer theory to be used that states that the analysis, $\boldsymbol{x}^a$, is simply the background, $\boldsymbol{x}^b$, plus a correction term [29]. The correction term is assumed to be a linear combination of the representer functions, $\boldsymbol{\gamma}_k$, with the coefficients, $\boldsymbol{\beta}_k$ for $k = 1, \ldots, K$. This implies that we can define the solution to the coupled linear Euler-Lagrange system as

$$\boldsymbol{x}^a - \boldsymbol{x}^b = \sum_{k=1}^{K} \boldsymbol{\gamma}_k\boldsymbol{\beta}_k = \mathbf{P}^b\mathbf{H}^T\boldsymbol{\beta}, \tag{18.38}$$

where the coupling coefficient, $\boldsymbol{\beta}$, is

$$\boldsymbol{\beta} = [\mathbf{H}\mathbf{P}^b\mathbf{H}^T + \mathbf{R}]^{-1}[\boldsymbol{y} - \mathbf{H}(\boldsymbol{x}^b)]. \tag{18.39}$$

This leads to the final solution as

$$\boldsymbol{x}^a = \boldsymbol{x}^b + \mathbf{P}^b\mathbf{H}^T[\mathbf{H}\mathbf{P}^b\mathbf{H}^T + \mathbf{R}]^{-1}[\boldsymbol{y} - \mathbf{H}(\boldsymbol{x}^b)]. \tag{18.40}$$

However, the theory that we have presented so far is for the linear model and observation operators, which is not going to be true for a lot of different geophysical systems. In [369] the authors present the theory to implement the representer method for the nonlinear case. The technique is to recast the coupled nonlinear Euler-Lagrange equations in (18.25) and (18.26) as a sequence of coupled problems, where each iteration step is linearized about an analyzed basic state/trajectory from the previous iteration.

In [369] the authors start with $\boldsymbol{x}^j$ and $\boldsymbol{x}^{j-1}$ being the estimated states from the jth and $(j-1)$th iteration, respectively. We now assume that the two iterations are linked through $\boldsymbol{x}^j = \boldsymbol{x}^{j-1} + \delta\boldsymbol{x}^j$. Next we linearize the nonlinear model $\boldsymbol{\mathcal{M}}$ and the nonlinear observation operator $\boldsymbol{\mathcal{H}}$ at the jth iteration around the previous analysis $(j-1)$ through a Taylor series expansion, such that

$$\begin{aligned}
&\boldsymbol{\lambda}_n^j - [\mathbf{M}_n^{j-1}]^T \boldsymbol{\lambda}_{n+1}^j = [\mathbf{H}_n^{j-1}]^T \sum_{n'=0}^{N} \mathbf{R}_{nn'}^{-1}(\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'}^{j-1}) - \mathbf{H}_{n'}^{j-1}\delta\boldsymbol{x}_{n'}^j), \\
&\quad \text{for } 1 \le n \le N-1, \\
&\quad \text{subject to} \quad \boldsymbol{\lambda}_N^j = [\mathbf{H}_N^{j-1}]^T \sum_{n'}^{N} \mathbf{R}_{Nn'}^{-1}(\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'}^{j-1}) - \mathbf{H}_{n'}^{j-1}\delta\boldsymbol{x}_{n'}^j),
\end{aligned} \tag{18.41}$$

and

$$\begin{aligned}
&\boldsymbol{x}_n^j - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1}^{j-1}) - \mathbf{M}_{n-1}^{j-1}\delta\boldsymbol{x}_{n-1}^j = 0, \quad \text{for } 1 \le n \le N, \\
&\quad \text{subject to} \\
&(\boldsymbol{x}_0^j - \boldsymbol{x}_0^b) = \mathbf{P}_0^b([\mathbf{M}_0^{j-1}]^T \boldsymbol{\lambda}_1^j + [\mathbf{H}_0^{j-1}]^T \sum_{n'=0}^{N} \mathbf{R}_{0n'}^{-1}(\boldsymbol{y}_{n'} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_{n'}^{j-1}) - \mathbf{H}_{n'}^{j-1}\delta\boldsymbol{x}_{n'}^j)).
\end{aligned} \tag{18.42}$$

18.2.2 Representer Solution to a Coupled Linearized Euler-Lagrange System

While (18.41) and (18.42) look similar to the linear coupled system (18.28) and (18.29), there are some subtle differences, and as such we briefly summarize the solution method with representers in this section. We make the same assumptions about non-temporal correlations and not using the observations at $n = 0$, as they have been assimilated in the previous window. This results in

$$\begin{aligned}
&\boldsymbol{\lambda}_n^j - [\mathbf{M}_n^{j-1}]^T \boldsymbol{\lambda}_{n+1}^j = [\mathbf{H}_n^{j-1}]^T \mathbf{R}_n^{-1}(\boldsymbol{y}_n - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_n^{j-1}) - \mathbf{H}_n^{j-1}\delta\boldsymbol{x}_n^j), \quad \text{for } 1 \le n \le N-1, \\
&\quad \text{subject to} \quad \boldsymbol{\lambda}_N^j = [\mathbf{H}_N^{j-1}]^T \mathbf{R}_N^{-1}(\boldsymbol{y}_N - \boldsymbol{\mathcal{H}}(\boldsymbol{x}_N^{j-1}) - \mathbf{H}_N^{j-1}\delta\boldsymbol{x}_N^j),
\end{aligned} \tag{18.43}$$

and

$$\begin{aligned}
&\boldsymbol{x}_n^j - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1}^{j-1}) - \mathbf{M}_{n-1}^{j-1}(\boldsymbol{x}_{n-1}^j - \boldsymbol{x}_{n-1}^{j-1}) = 0, \quad \text{for } 1 \le n \le N, \\
&\quad \text{subject to} \quad \boldsymbol{x}_0^j = \boldsymbol{x}_0^b + \mathbf{P}_0^b[\mathbf{M}_0^{j-1}]^T \boldsymbol{\lambda}_1^j.
\end{aligned} \tag{18.44}$$

The next step is to introduce a prior linear state, denoted as $(\boldsymbol{x}^p)^j$ for the jth iteration, that satisfies the following governing equation that has been linearized about the previous analysis $\boldsymbol{x}^{j-1}$:

$$\begin{aligned}
&(\boldsymbol{x}^p)^j - \boldsymbol{\mathcal{M}}(\boldsymbol{x}_{n-1}^{j-1}) - \mathbf{M}_{n-1}^{j-1}((\boldsymbol{x}^p)_{n-1}^j - \boldsymbol{x}_{n-1}^{j-1}) = \mathbf{0}, \quad \text{for } 1 \le n \le N, \\
&\quad \text{subject to} \quad (\boldsymbol{x}^p)_0^j = \boldsymbol{x}_0^b.
\end{aligned}$$

Next we introduce two new increments as

$$\delta \boldsymbol{x}_n^j = \boldsymbol{x}_n^j - \boldsymbol{x}_n^{j-1} \quad \text{and} \quad (\boldsymbol{x}^p)_n^j = (\boldsymbol{x}^p)_n^j - \boldsymbol{x}_n^{j-1}, \tag{18.45}$$

where $(\boldsymbol{x}_n^p)^j$ is used as the background state for the minimization of the linear, inner loop, problem. The analysis, $\boldsymbol{x}^j$, for the jth iteration can now be written as

$$\boldsymbol{x}^j - (\boldsymbol{x}^p)^j = (\mathbf{P}^b)^{j-1}[\mathbf{H}^{j-1}]^T \boldsymbol{\beta}^j \quad \text{or} \quad \delta \boldsymbol{x}^j - \delta(\boldsymbol{x}^p)^j = (\mathbf{P}^b)^{j-1}[\mathbf{H}^{j-1}]^T \boldsymbol{\beta}^j, \tag{18.46}$$

where the $\boldsymbol{\beta}^j$ are the coupling vectors or the analysis residual normalized by the observation error covariance for the jth iteration [369] and satisfy the following equation:

$$\boldsymbol{\beta}^j [\mathbf{H}^{j-1}(\mathbf{P}^b)^{j-1}[\mathbf{H}^{j-1}]^T + \mathbf{R}]^{-1}(\boldsymbol{y} - \boldsymbol{\mathcal{H}}(\boldsymbol{x}^{j-1}) - \mathbf{H}^{j-1}\delta(\boldsymbol{x}^p)^j). \tag{18.47}$$

For the derivation of (18.47) please refer to Appendix in [369].

The details about implementing the accelerated representer (AR) method are presented in [489], and we shall only briefly summarize them here, while the explanation in [489] is for the linear case, where we would simply replace $\boldsymbol{y}$ with $\mathbf{d} = \boldsymbol{y} - \boldsymbol{\mathcal{H}}(\boldsymbol{x})$. Their description of the implementation of the AR starts with the postmultiplication step $\mathbf{P}^b\mathbf{H}^T\mathbf{z}$:

1. Let $\mathbf{z} = [\mathbf{z}_1^T \ldots \mathbf{z}_n^T \ldots \mathbf{z}_N^T]$ where $\mathbf{z}_n^T \in \mathbb{R}^{k_n}$ for time t_n. Let $\mathbf{f}_n$ and $\mathbf{g}_n \in \mathbb{R}^I$ for each t_n.
2. Calculate

$$\mathbf{f}_n = \mathbf{M}^T \mathbf{f}_{n+1} + \mathbf{H}_n^T \mathbf{z}_n, \quad \text{for } 0 \le n \le N-1, \quad \text{subject to } \mathbf{f}_N = \mathbf{H}_N^T \mathbf{z}_N, \tag{18.48}$$

   where we are using the adjoint of the forecast model. This step is the **backward sweep**, and the end result is the vector $\mathbf{f}_0$.
3. The next step is to apply the **forward sweep** using the tangent linear forecast model starting at time t_0 such that

$$\mathbf{g}_n = \mathbf{M}_{n-1}\mathbf{g}_{n-1} + \mathbf{Q}_n\mathbf{f}_n, \quad \text{for } 1 \le n \le N, \quad \text{subject to } \mathbf{g}_0 = \mathbf{P}_0^b\mathbf{f}_0. \tag{18.49}$$

4. Combining the steps above results in

$$\boldsymbol{x}_n^a = \boldsymbol{x}_n^b + \mathbf{g}_n, \quad \text{for } 0 \le n \le N. \tag{18.50}$$

We should note here that the $\mathbf{z}$ vectors have come out of the solver step, which according to [489] is similar to the post-multiplier, but involves using a conjugate gradient method to solve the set of linear equations of the system

$$(\mathbf{H}^{j-1}(\mathbf{P}^b)^{j-1}[\mathbf{H}^{j-1}]^T + \mathbf{R})^{-1}\mathbf{z} = \mathbf{d} - \mathbf{H}\delta(\boldsymbol{x}^p)^j. \tag{18.51}$$

Another feature to note about the algorithm above is that from (18.50) we see that the analysis state can be updated anywhere in the window, as we have $\mathbf{g}_n$ for all n, not just for $n = 0$.

18.3 Duality of the VAR and PSAS Systems

If we recall the concept of duality from control theory, then if we are able to form the problem in control space or we could form the problem in observation space, and as long as one was completely controllable in control space, then the projected version in observation space would be completely observable, and vice versa. In a paper by Dr. Philippe Courtier in 1997, he was able to link the model space 3D and 4D VAR systems to their equivalent PSAS systems in observation space [75].

In this section we shall briefly outline his proof of the duality between the two systems. When dealing with data assimilation, scientists refer to the observation-based space VAR systems as the **dual** problem, or as being in the dual space, rather than PSAS.

We start with the incremental 3D VAR cost function

$$J(\delta \boldsymbol{x}) = \frac{1}{2}\delta \boldsymbol{x}\mathbf{B}^{-1}\delta \boldsymbol{x} + \frac{1}{2}(\mathbf{d} - \mathbf{H}\delta \boldsymbol{x})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\delta \boldsymbol{x}).$$

We apply the preconditioner $\mathbf{B} = \sqrt{\mathbf{B}}\sqrt{\mathbf{B}}$, where the inverse is $\mathbf{B}^{-1} = \sqrt{\mathbf{B}^{-1}}\sqrt{\mathbf{B}^{-1}}$. Therefore, there are two spaces involved here: the model space and the observation space, where the analysis increments, $\delta \boldsymbol{x}^a$, are in model space while the innovations $\mathbf{d}$ are in observation space. These spaces are connected through the linearized observation operator $\mathbf{H}$, such that $\mathbf{H}\delta \boldsymbol{x}^a$ is the analysis increments in observation space.

Introducing the change of variable, $\mathbf{v} = \sqrt{\mathbf{B}^{-1}}\delta \boldsymbol{x}$, to precondition the problem [260], enables the 3D VAR cost function to become

$$J(\mathbf{v}) = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{H}\sqrt{\mathbf{B}}\mathbf{v})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\sqrt{\mathbf{B}}\mathbf{v}), \tag{18.52}$$

whose Jacobian with respect to $\mathbf{v}$ is

$$\nabla_{\mathbf{v}} J = (\mathbf{I} + \sqrt{\mathbf{B}}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\sqrt{\mathbf{B}})\mathbf{v} - \sqrt{\mathbf{B}}\mathbf{H}^T\mathbf{d}, \tag{18.53}$$

and the Hessian matrix is given by

$$\mathbf{J}'' = (\mathbf{I} + \sqrt{\mathbf{B}}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\sqrt{\mathbf{B}}). \tag{18.54}$$

In the end, as we are seeking the minimum of the cost function, we are solving the following linear system of equations for $\delta \boldsymbol{x}$:

$$(\mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})\delta \boldsymbol{x} = \mathbf{H}^T\mathbf{R}^{-1}\mathbf{d}. \tag{18.55}$$

For the PSAS situation, we have from [75] that through the series of change of variables, $\delta \boldsymbol{x} = \mathbf{B}\mathbf{H}^T\mathbf{w}$ and $\mathbf{u} = \sqrt{\mathbf{R}}\mathbf{w}$, the Hessian matrix for the PSAS formulation is given by

$$\mathbf{F}''_{\mathbf{u}} = \mathbf{I} + \sqrt{\mathbf{R}^{-1}}\mathbf{H}\mathbf{B}\mathbf{H}^T\sqrt{\mathbf{R}^{-1}}, \tag{18.56}$$

where F is the cost function

$$F(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T(\mathbf{I} + \sqrt{\mathbf{R}^{-1}}\mathbf{H}\mathbf{B}\mathbf{H}^T\sqrt{\mathbf{R}^{-1}})\mathbf{u} - \mathbf{u}^T\sqrt{\mathbf{R}^{-1}}\mathbf{d}. \tag{18.57}$$

The duality of the two approaches comes about through showing that the preconditioned Hessian matrices for the two formulations have the same condition number. We have already seen that the model space and the observation space are connected via the linearized observation operator $\mathbf{H}$, and that we have the property

$$\mathbf{H}^T\sqrt{\mathbf{R}^{-1}}\mathbf{u} = \sqrt{\mathbf{B}^{-1}}\mathbf{v} = \mathbf{B}^{-1}\delta\boldsymbol{x}. \tag{18.58}$$

The proof of duality between the 3D VAR and PSAS, the 4D VAR proof, can be found in [75] as it involves the sweeping that we described above and that in the observation space we are creating analysis increments at each observation time [75]. However, we shall state the two lemmas from [75] that prove duality between the two approaches.

Lemma 18.1. *The two matrices* $\sqrt{\mathbf{R}^{-1}}\mathbf{H}\mathbf{B}\mathbf{H}^T\sqrt{R^{-1}}$ *and* $\sqrt{\mathbf{H}\mathbf{B}\mathbf{H}^T}\mathbf{R}^{-1}\sqrt{\mathbf{H}\mathbf{B}\mathbf{H}^T}$ *have the same eigen-spectrum.*

Lemma 18.2. *The two matrices* $\sqrt{\mathbf{H}\mathbf{B}\mathbf{H}^T}\mathbf{R}^{-1}\sqrt{\mathbf{H}\mathbf{B}\mathbf{H}^T}$ *and* $\sqrt{\mathbf{B}}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\sqrt{B}$ *have the same eigen-spectrum.*

Therefore, by showing that the two Hessian matrices have the same spectrum, this implies that they have the same condition number and as such, according to [75], they are solving the same problem.

18.4 Summary

In this chapter we have introduced an alternative formulation for the variational methods, but this time the problem is solved in observation space. It is quite often the case that there are many magnitude of difference between the total number of observations and the total number of grid points or spectral coefficients. As the number of observations grows, the advantage of using the observation space approach may not be as good; however, as the resolution of the numerical models increase and extra analysis variables are needed to be minimized for, then the area that has the slowest growth in dimensionality may still have the advantage.

The method of ARs is used by the United States' Navy for atmospheric data assimilation as well as for an ocean assimilation system. Representers, however, have been used in ocean data assimilation [29,62], in shallow water modeling on with a cubed-sphere grid [202] as well as in many hydrological applications. The PSAS-based system has also been applied in a chemical state analysis [177].

We leave the variational methods for now, until the non-Gaussian chapter, and move on to consider an alternative form of data assimilation. As we have stated, the variational approaches seek the maximum likelihood state, or mode, while the OI approach seeks the minimum variance state but in local areas or at grid points. We now consider the techniques to seek the minimum variance estimates, but without localization. This method is of course the **Kalman filter**.

This page intentionally left blank

CHAPTER

Kalman Filter and Smoother 19

Contents

The discrete version of the Kalman filter first appeared in Kalman's 1960 paper: *A new approach to linear filtering and prediction problems* [205]. The basis of his work was related to problems in communications and control that are of a statistical nature [205]. The class of problems that Kalman is referring to are: (1) prediction of random signals; (2) separation of random signals from random noise; and (3) detection of signals of known forms in the presence of random noise.

Kalman makes reference to the work of Wiener [475], and says that Wiener's method to approach problems (1) and (2) above leads to what is known as the **Wiener-Hopf** integral equation. We shall not go into details about this problem here, but we note that it is part of the motivation for Kalman.

Kalman lists a series of paper that have proposed methods to solve the Wiener-Hopf integral equations as well as many different generalizations, where the objective of these series of papers was to obtain specifications of a linear dynamic system, which Kalman refers to as the **Wiener filter**, that accomplishes the prediction, separation, or detection of a random signal.

Kalman then states four problems for this method, which he shows do not afflict his new filter:

1. The optimal filter is specified by its impulse response.
2. Numerical determination of the optimal impulse response is often quite involved and poorly suited for machine computation.
3. Important generalizations require new derivations, frequently of considerable difficulty.
4. The mathematics of the derivation are not transparent.

However, Kalman says that his new approach produces the following:

1. **Optimal estimates and orthogonal projections**. His approach is to consider the Wiener filter in terms of conditional distributions and expectations. He shows that all of the statistical calculations and results are based upon first- and second-order averages; **no other statistical data are required**.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00023-1

2. **Models for random processes**. In Kalman's approach, arbitrary random signals are represented as the output of a linear dynamic system that is excited by independent random signals.
3. **The solution of the Wiener problem**. Kalman uses the state-transition method, where, as a result of this approach, the single derivation then covers a large variate of problems. **Guessing** the *state* of the estimation correctly leads to a nonlinear difference/differential equation for the covariance matrix of the optimal estimation error. The solution of the equation for the covariance matrix starts at time t_0, which is where the first observation is taken; at each later time t the solution of the equation represents the covariance of the optimal prediction error, given observations in the interval (t_0, t). Kalman then states that

> From the covariance matrix at time t we obtain at once, without further calculations, the coefficients, which could be time-varying, characterizing the optimal linear filter.

Given Kalman's motivation above we now move on to the derivation of the Kalman filter as set out in [205] and then show a direct statistical-based derivation.

19.1 Derivation of the Kalman Filter

The initial motivation for the derivation of the Kalman filter equations in [205] start with an example: suppose that we are given a signal $x_1(t)$ and noise $x_2(t)$; however, only the sum of the two $y(t) = x_1(t) + x_2(t)$ can be observed. Next, suppose that we have observed and know exactly the values $y(t_0), \ldots, y(t)$. Now there are three possible situations with regard to the knowledge of the unobservable value of the signal at $t = t_1$:

1. If $t_1 < t$, this is a **data smoothing (interpolation)** problem.
2. If $t_1 = t$, this is a **filtering** problem.
3. If $t_1 > t$, this is a **prediction** problem.

So we now see where the terms "smoothing" and "filtering" come from [205]. It is stated in [205] that Kalman is considering all three cases and refers to them collectively as **estimation**.

Now to the mathematics and probability. We assume that the signal, noise, and their sum are random processes, where we can determine the probability with which a particular sample of the signal and noise will occur. Therefore, for any given set of measured values $\mathbf{y}(t_0), \ldots, \mathbf{y}(t)$ of the random variable $y(t)$, it is possible to determine the probability of the simultaneous occurrence of various values, $\mathbf{x}_1(t)$, of the random variable $x_1(t_1)$. This leads to the conditional probability of

$$P(x_1(t_1) \leq \mathbf{x}_1 | y(t_0)) = \mathbf{y}(t_0), \ldots, y(t) = \mathbf{y}(t) = F(\mathbf{x}_1). \tag{19.1}$$

The function $F(\mathbf{x}_1)$ represents all the information that the measurement of the random variables, $y(t_0), \ldots, y(t)$, have provided about the random variable $x_1(t_1)$ and is a conditional PDF.

We now denote a statistical estimate of the random variable, $x_1(t_1)$, as $X_1(t_1|t) \equiv X_1(t_1) \equiv X_1$. To be able to arrive at a way of determining $x_1(t_1)$, Kalman introduces a **penalty or loss function**, L, that should be positive and non-decreasing function of the **estimation error**, $\boldsymbol{\varepsilon} = x_1(t_1) - X_1(t_1)$, and the

function should have the following three properties:

$$L(0)=0, \quad L(\boldsymbol{\varepsilon}_1)\geq L(\boldsymbol{\varepsilon}_2)\geq 0 \;\; (\boldsymbol{\varepsilon}_2\geq\boldsymbol{\varepsilon}_1\geq 0), \quad L(\boldsymbol{\varepsilon})=L(-\boldsymbol{\varepsilon}). \tag{19.2}$$

The choice for the loss function that Kalman selects is the one that *minimized the average loss or risk*, which is defined as

$$\mathbb{E}[L(x_1(t_1)-X_1(t_1))]=\mathbb{E}[\mathbb{E}[L(x_1(t_1)-X_1(t_1))|y(t_0),\ldots,y(t)]]. \tag{19.3}$$

However, we should note that we can remove the first expectation operation on the right-hand side of (19.3) as it is not operating on X_1, and such we have

$$\mathbb{E}[L(x_1(t_1)-X_1(t_1))|y(t_0),\ldots,y(t)]. \tag{19.4}$$

Given these assumptions, Kalman introduces the following very important theorem.

Theorem 19.1. *Given a loss function that satisfies the conditions in* (19.2) *and that the conditional distribution function* $F(\mathbf{x})$*, defined by* (19.1)*, is symmetric about its mean,* $F(\mathbf{x}-\overline{\mathbf{x}})=1-F(\overline{\mathbf{x}}-\mathbf{x})$*, and is also convex*

$$F(\lambda\mathbf{x}_1+(1-\lambda)\mathbf{x}_2)\leq\lambda F(\mathbf{x}_1)+(1-\lambda)F(\mathbf{x}_2),$$

for all $\mathbf{x}_1$, $\mathbf{x}_2\leq\overline{\mathbf{x}}$ *and for* $0\leq\lambda\leq 1$ *then the random variable* $\mathbf{x}_1^*(t_1|t)$ *that minimizes the average loss given by* (19.3) *is the conditional expectation*

$$x_1^*(t_1|t)=\mathbb{E}[x_1(t_1)|y(t_0),\ldots,y(t)]. \tag{19.5}$$

Orthogonal projections

Given the real valued random variables $y(t_0),\ldots,y(t)$, then the set of all linear combinations of these random variables with real coefficients

$$\sum_{i=t_0}^{t} a_i y(i), \tag{19.6}$$

form a vector space/linear manifold, $\mathcal{Y}(t)$, where this space should be considered as a finite dimensional subspace of the space of all possible observations. It is possible to select an orthonormal basis in $\mathcal{Y}(t)$, such that the set of vectors $\mathbf{e}_{t_0}\ldots\mathbf{e}_t$ in $\mathcal{Y}(t)$ enables any vector in $\mathcal{Y}(t)$ to be expressed as a unique linear combination of these vectors such that

$$\mathbb{E}\left[\mathbf{e}_i,\mathbf{e}_j\right]=\delta_{i,j}, \tag{19.7}$$

where δ is a delta function that is equal to 1 when $i=j$ and equal to 0 when $i\neq j$. Therefore, any vector, $\overline{\mathbf{x}}$ in $\mathcal{Y}(t)$ can be written as

$$\overline{\mathbf{x}}=\sum_{i=t_0}^{t} a_i\mathbf{e}_i, \tag{19.8}$$

where the coefficients a_i are determined by

$$\mathbb{E}\left[\overline{\mathbf{x}}\mathbf{e}_j\right] = \mathbb{E}\left[\left(\sum_{i=t_0}^{t} a_i \mathbf{e}_i\right)\mathbf{e}_j\right] = \sum_{i=t_0}^{t} a_i \mathbb{E}\left[\mathbf{e}_i \mathbf{e}_j\right] = \sum_{i=t_0}^{t} a_i \delta_{ij} = a_j. \tag{19.9}$$

Thus any random variable $\mathbf{x}$, that may not be observed, can be uniquely decomposed into two parts: $\overline{\mathbf{x}} \in \mathcal{Y}(t)$ and a component, $\tilde{\mathbf{x}}$, which is orthogonal to $\mathcal{Y}(t)$. The component $\overline{\mathbf{x}}$ is referred to as the orthogonal projection of $\mathbf{x}$ on $\mathcal{Y}(t)$.

Given these projections, and some other properties that Kalman presents in [205], we arrive at the second important theorem:

Theorem 19.2. *Let $\{x(t)\}$ $\{y(t)\}$ be random processes with zero mean. We observe $y(t_0), \ldots, y(t)$ if either the random processes are Gaussian, or the optimal estimate is restricted to be a linear function of the observed random variables and the loss function is $L(\boldsymbol{\varepsilon}) = \boldsymbol{\varepsilon}^2$; the $x^*(t_1|t)$ is the optimal estimate of $x(t_1)$ given $y(t_0), \ldots, y(t)$. It is also the orthogonal projection $\overline{\mathbf{x}}(t_1|t)$ of $\mathbf{x}(t_1)$ on $\mathcal{Y}(t)$.*

The interpretation of this theorem is that the optimal estimate, given the two conditions in Theorem 19.2, is a linear combination of all previous observations, and can be regarded as the output of a linear filter, with the input being the actually occurring values of the observable random variables.

The next step in the derivation of the Kalman filter is to determine the models for the random processes, and this is where the control theory aspect comes in. Kalman decides that

> A random function of time may be thought of as the output of a dynamic system excited by an independent Gaussian random process.

Kalman now makes the statement that a Gaussian random signal remains Gaussian after passing through a linear system. Therefore if we assume independent Gaussian primary random sources, and if the observed random signal is also Gaussian, then we have to assume that the dynamic system between the observer and the primary source is **linear**.

Given these assumptions we now introduce the control system

$$\left.\begin{aligned} \dot{\mathbf{x}} &= \mathbf{M}(t)\,\mathbf{x} + \mathbf{B}(t)\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{H}(t)\,\mathbf{x}(t), \end{aligned}\right\} \tag{19.10}$$

as we saw in Chapter 6. If we now assume a time-invariant case where the matrices in (19.10) are constant and that the control $\mathbf{u}$ is constant as well, then we can write the solution at $t+1$ as

$$\mathbf{x}(t+1) = \boldsymbol{\Phi}(1)\,\mathbf{x}(t) + \boldsymbol{\Delta}\mathbf{u}(t); \quad t = 0, 1, \ldots, \tag{19.11}$$

where $\boldsymbol{\Phi}$ is the state transition matrix as defined earlier and is given here by

$$\boldsymbol{\Phi}(1) = e^{\mathbf{M}} \equiv \sum_{i=0}^{\infty} \frac{\mathbf{M}^i}{i!} \tag{19.12}$$

and

$$\boldsymbol{\Delta}(1) = \left(\int_0^1 e^{\mathbf{M}\tau} d\tau\right)\mathbf{B}. \tag{19.13}$$

Given the introduction of the linear dynamic system, Kalman decides upon the following model to be the basis for the filter:

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{\Phi}(t+1,t)\,\mathbf{x}(t) + \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{H}(t)\,\mathbf{x}(t), \end{aligned} \tag{19.14}$$

where $\mathbf{u}(t)$ is an independent Gaussian random process of *n-vectors* with zero mean, where the state transition matrix and the observation operator are non-random functions of time. We should note here that the there is no error associated with the observations, therefore this is no $\mathbf{R}$ matrix. In fact the measurement error is introduced in the Kalman and Bucy (1961) paper [206].

Therefore, given the observed values of $\mathbf{y}(t_0), \ldots, \mathbf{y}(t)$, we need to find an estimate, $\mathbf{x}^*(t_1|t)$, of $\mathbf{x}(t_1)$, that minimizes the expected loss.

The problem, as just defined, includes the three cases of filtering, smoothing, and prediction. It also includes the problem of reconstructing all the state variables of a linear dynamic system from noisy observations of some of the state variables.

We assume that $\mathbf{y}(t_0) \ldots \mathbf{y}(t-1)$ have been measured. Next, at time t the random variable $\mathbf{y}(t)$ is measured, and we have that $\widetilde{\mathbf{y}}(t|t-1)$ is the component of $\mathbf{y}(t)$ that is orthogonal to $\mathcal{Y}(t-1)$, that generates a linear manifold, $\mathcal{Z}(t)$.

Now we assume, by induction, that $\mathbf{x}^*(t_1-1|t-1)$ is known; thus we can define

$$\begin{aligned} \mathbf{x}^*(t_1|t) &= \mathbb{E}[\mathbf{x}(t_1)\,|\mathcal{Y}(t)] = \mathbb{E}[\mathbf{x}(t_1)\,|\mathcal{Y}(t-1)] + \mathbb{E}[\mathbf{x}(t_1)\,|\mathcal{Z}(t)], \\ &= \mathbf{\Phi}(t+1,t)\,\mathbf{x}^*(t_1-1|t-1) + \mathbb{E}[\mathbf{u}(t_1-1)\,|\mathcal{Y}(t-1)] + \mathbb{E}[\mathbf{x}(t_1)\,|\mathcal{Z}(t)]. \end{aligned} \tag{19.15}$$

Through the orthogonality property, Kalman is able to remove the term involving the control in (19.15) and as such, given the requirement that everything is Gaussian if the function is linear, then we require the third term on the right-hand side of (19.15) to be linear. Therefore, we have

$$\mathbb{E}[\mathbf{x}(t+1)\,|\mathcal{Z}(t)] = \mathbf{\Delta}^*\widetilde{\mathbf{y}}(t|t-1), \tag{19.16}$$

where $\mathbf{\Delta}^*(t)$ is a rectangular matrix, and the star refers to *optimal filtering*.

From the work on the manifold, we know that the component of $\mathbf{y}(t)$ that lies on the manifold $\mathcal{Y}(t-1)$ is $\overline{\mathbf{y}}(t|t-1) = \mathbf{H}(t)\,\mathbf{x}^*(t|t-1)$. This implies that

$$\widetilde{\mathbf{y}}(t|t-1) = \mathbf{y}(t) - \overline{\mathbf{y}}(t|t-1) = \mathbf{y}(t) - \mathbf{H}(t)\,\mathbf{x}^*(t|t-1). \tag{19.17}$$

Combining (19.15)–(19.17) and factorizing the $\mathbf{x}^*(t|t-1)$ terms, results in

$$\mathbf{x}^*(t+1|t) = \mathbf{\Phi}^*(t+1,t)\,\mathbf{x}^*(t|t-1) + \mathbf{\Delta}^*(t)\,\mathbf{y}(t), \tag{19.18}$$

where

$$\mathbf{\Phi}^*(t+1,t) = \mathbf{\Phi}(t+1,t) - \mathbf{\Delta}^*\mathbf{H}(t). \tag{19.19}$$

After all this work, we can say that the optimal estimation is performed by a linear dynamic system of the same form as (19.14). The state of the estimator is the previous estimate, the input is the last measured value of the observable random variable $\mathbf{y}(t)$, and the transition matrix is given by (19.19).

The estimation error $\widetilde{\mathbf{x}}(t+1|t)$ is also governed by a linear dynamic system and there we have

$$\begin{aligned}\widetilde{\mathbf{x}}(t+1|t) &= \mathbf{x}(t+1) - \mathbf{x}^*(t+1|t),\\ &= \boldsymbol{\Phi}(t+1,t)\,\mathbf{x}(t) + \mathbf{u}(t) - \boldsymbol{\Phi}^*(t+1,t)\,\mathbf{x}^*(t|t-1) - \boldsymbol{\Delta}^*(t)\,\mathbf{H}(t)\,\mathbf{x}(t),\\ &= \boldsymbol{\Phi}^*(t+1,t)\,\widetilde{\mathbf{x}}(t|t-1) + \mathbf{u}(t).\end{aligned} \tag{19.20}$$

We can see that (19.20) is a recursive relationship for the covariance matrix $\mathbf{P}^a(t)$ of the optimal error $\widetilde{\mathbf{x}}(t|t-1)$. Note that we are using the notation from the convention set out in [190]. Therefore, here we shall derive the forecast error covariance matrix from (19.20):

$$\begin{aligned}\mathbf{P}^f(t+1) &= \mathbb{E}\left[\widetilde{\mathbf{x}}(t+1|t)\,\widetilde{\mathbf{x}}^T(t+1|t)\right],\\ &= \boldsymbol{\Phi}^*(t+1,t)\,\mathbb{E}\left[\widetilde{\mathbf{x}}(t|t-1)\,\widetilde{\mathbf{x}}^T(t|t-1)\right]\boldsymbol{\Phi}^{*T}(t+1,t) + \mathbf{Q}(t),\\ &= \boldsymbol{\Phi}^*(t+1,t)\,\mathbb{E}\left[\widetilde{\mathbf{x}}(t|t-1)\,\widetilde{\mathbf{x}}^T(t|t-1)\right]\boldsymbol{\Phi}^{T}(t+1,t) + \mathbf{Q}(t),\\ &= \boldsymbol{\Phi}^*(t+1,t)\,\mathbf{P}^f(t)\,\boldsymbol{\Phi}^T(t+1,t) + \mathbf{Q}(t),\end{aligned} \tag{19.21}$$

where $\mathbf{Q}(t) \equiv \mathbb{E}\left[\mathbf{u}(t)\,\mathbf{u}^T(t)\right]$.

We still require an expression for $\boldsymbol{\Delta}^*$, along with $\boldsymbol{\Phi}^*$, since

$$\widetilde{\mathbf{x}}(t+1)\,|Z(t) = \mathbf{x}(t+1) - \mathbf{E}\left[\mathbf{x}(t+1)\,|\boldsymbol{\mathcal{Z}}(t)\right],$$

is orthogonal to $\widetilde{\mathbf{y}}(t|t-1)$, then it follows from (19.16) that

$$\begin{aligned}\mathbf{0} &= \mathbb{E}\left[\left(\mathbf{x}(t+1) - \boldsymbol{\Delta}^*(t)\,\widetilde{\mathbf{y}}(t|t-1)\right)\widetilde{\mathbf{y}}^T(t|t-1)\right],\\ &= \mathbb{E}\left[\mathbf{x}(t+1)\,\widetilde{\mathbf{y}}^T(t|t-1)\right] - \boldsymbol{\Delta}^*(t)\,\mathbb{E}\left[\widetilde{\mathbf{y}}(t|t-1)\,\widetilde{\mathbf{y}}^T(t|t-1)\right].\end{aligned} \tag{19.22}$$

An important feature to note here is that Kalman substitutes $\mathbf{y} = \mathbf{H}\mathbf{x}$ which does **not** have an observational error as we mentioned before; however, when you see the Kalman filter equations in the literature, there is an $\mathbf{R}$ matrix. Therefore, in the 1960 paper there is no observational error covariance matrix. We shall add it here and use the fact that we can drop the control term due to the orthogonality to $\boldsymbol{\mathcal{Z}}$ which implies that

$$\begin{aligned}\mathbb{E}\left[\widetilde{\mathbf{y}}(t|t-1)\,\widetilde{\mathbf{y}}^T(t|t-1)\right] &\equiv \mathbb{E}\left[\left(\mathbf{H}(t)\,\widetilde{\mathbf{x}}(t|t-1) + \boldsymbol{\varepsilon}^o\right)\left(\mathbf{H}(t)\,\widetilde{\mathbf{x}}(t|t-1) + \boldsymbol{\varepsilon}^o\right)^T\right],\\ &= \mathbf{H}(t)\,\mathbf{P}^f(t)\,\mathbf{H}^T(t) + \mathbf{R}(t).\end{aligned} \tag{19.23}$$

After some substitution for known expectations in (19.22), we obtain

$$\boldsymbol{\Delta}^*(t) = \boldsymbol{\Phi}(t+1,t)\,\mathbf{P}^f(t)\,\mathbf{H}^T(t)\left(\mathbf{H}(t)\,\mathbf{P}^f(t)\,\mathbf{H}(t) + \mathbf{R}(t)\right)^{-1}. \tag{19.24}$$

The expression multiplying the state transition matrix should look familiar; it is the gain matrix, as we saw with the optimum interpolating derivation. This matrix is often denoted as $\mathbf{K}$ and is called the **Kalman gain matrix**; we shall define it as $\mathbf{K} \equiv \mathbf{P}^f(t)\,\mathbf{H}^T(t)\left(\mathbf{H}(t)\,\mathbf{P}^f(t)\,\mathbf{H}(t) + \mathbf{R}(t)\right)^{-1}$.

After all this work, we are now in a position to summarize all of these results into the following theorem from [205].

Theorem 19.3. *Given the linear dynamic system in (19.14), then the optimal estimate* $\mathbf{x}^*(t+1|t)$ *of* $\mathbf{x}(t+1)$*, given the set of observations, is generated by the linear dynamic system*

$$\mathbf{x}^*(t+1|t) = \mathbf{\Phi}^*(t+1,t)\,\mathbf{x}^*(t|t-1) + \mathbf{\Delta}^*(t)\,\mathbf{y}(t). \tag{19.25}$$

The estimation (forecast) error is given by

$$\widetilde{\mathbf{x}}(t+1|t) = \mathbf{\Phi}^*(t+1,t)\,\widetilde{\mathbf{x}}(t|t-1) + \mathbf{u}(t). \tag{19.26}$$

The forecast error covariance matrix is given by

$$\mathbf{P}^f(t) = \mathbf{\Phi}^*(t,t-1)\,\mathbf{P}^f(t-1)\,\mathbf{\Phi}^T(t,t-1) + \mathbf{Q}(t-1). \tag{19.27}$$

The expected quadratic loss is

$$\sum_{i=1}^{n} \mathbb{E}\left[\tilde{x}_i^2(t|t-1)\right] = Trace\left\{\mathbf{P}^*(t)\right\}. \tag{19.28}$$

Finally we have the Kalman gain matrix and the update step as

$$\mathbf{\Delta}^*(t) = \mathbf{\Phi}(t+1,t)\,\mathbf{P}^f(t)\,\mathbf{H}^T(t)\left(\mathbf{H}(t)\,\mathbf{P}^f(t)\,\mathbf{H}^T(t) + \boldsymbol{R}\right)^{-1}, \tag{19.29}$$

$$\mathbf{\Phi}^*(t+1) = \mathbf{\Phi}(t+1,t) - \mathbf{\Delta}^*(t)\,\mathbf{H}(t), \tag{19.30}$$

$$\mathbf{P}^{\mathbf{f}}(t+1) = \mathbf{\Phi}(t+1,t)\,(\mathbf{I} - \mathbf{KH})\,\mathbf{\Phi}^T(t+1,t) + \mathbf{Q}(t). \tag{19.31}$$

That is a very daunting, and not very friendly, summary of the original derivation of the Kalman filter from [205], where we note that we had to add the $\mathcal{E}\left[\mathbf{y}\mathbf{y}^t\right] = \mathbf{R}$ term, which is missing in the 1960 paper, but is present in the derivation for the continuous filter derived in the Kalman and Bucy (1961) paper. We now present a derivation of the Kalman filter from a more statistical direction.

19.2 Kalman Filter Derivation From a Statistical Approach

There is an easier way to derive the Kalman filter equations, which involves following Kalman's framework, but without the manifolds and state transition matrices. We start with the usual concept of the background state, which is the forecast from the previous analysis state as

$$\mathbf{x}^b_{t|t-1} \equiv \mathbf{M}_{t,t-1}\mathbf{x}^{\mathbf{a}}{}_{t-1|t-1}. \tag{19.32}$$

Let $\mathbf{x}^t_t$ be the true state at time t, then the background/analysis error is given by

$$\boldsymbol{\varepsilon}^a_t = \mathbf{x}^b_{t|t-1} - \mathbf{x}^t_{t|t} = \mathbf{M}_{t,t-1}\mathbf{x}^{\mathbf{a}}{}_{t-1|t-1} - \mathbf{x}^t_{t|t}. \tag{19.33}$$

We know that the analysis at the previous filtering time has an associated analysis error such that

$$\mathbf{x}^a_{t-1|t-1} = \mathbf{x}^t_{t-1|t-1} + \boldsymbol{\varepsilon}^a_{t-1|t-1}. \tag{19.34}$$

This means we can write the forecast/background error as

$$\begin{aligned}\boldsymbol{\varepsilon}_t^b &= \mathbf{M}\mathbf{x}_{t-1|t-1}^t + \mathbf{M}\boldsymbol{\varepsilon}_{t-1}^a - \mathbf{x}_{t|t}^t, \\ &= \mathbf{M}\boldsymbol{\varepsilon}_{t-1}^a + \boldsymbol{\varepsilon}_t^m,\end{aligned} \tag{19.35}$$

where $\boldsymbol{\varepsilon}^m$ is the model error given by $\boldsymbol{\varepsilon}_t^m \equiv \mathbf{M}\mathbf{x}_{t-1|t-1}^t - \mathbf{x}_{t|t}^t$. If we now form the background/forecast error covariance matrix, we have

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{\varepsilon}_t^b, \left(\boldsymbol{\varepsilon}_t^b\right)^T\right] = \mathbf{P}_t^b &= \mathbb{E}\left[\left(\mathbf{M}\boldsymbol{\varepsilon}_{t-1|t-1}^a + \boldsymbol{\varepsilon}_t^m\right)\left(\mathbf{M}\boldsymbol{\varepsilon}_{t-1|t-1}^a + \boldsymbol{\varepsilon}_t^m\right)^T\right], \\ &= \mathbf{M}\mathbb{E}\left[\boldsymbol{\varepsilon}_{t-1|t-1}^a\left(\boldsymbol{\varepsilon}_{t-1|t-1}^a\right)^T\right]\mathbf{M}^T + \mathbb{E}\left[\boldsymbol{\varepsilon}_t^m, \boldsymbol{\varepsilon}_t^m\right], \\ &= \mathbf{M}\mathbf{P}_{t-1}^a\mathbf{M}^T + \mathbf{Q}_t.\end{aligned} \tag{19.36}$$

The expression in (19.36) is the same as for the derivation in the last section.

Now, given a predicted/forecasted state, $\mathbf{x}_{k+1|k}$, which is associated with observations up to time t_k, and assuming that we have received an observation of the state at $t = t_{k=1}$, we wish to obtain an estimate of the state at $t = t_{k+1}$, given the observation at $t = t_{k+1}$, i.e., $\mathbf{x}_{k+1|k+1}$. We assume that the estimate is a weighted sum of the prediction and the new observation, given by the equation

$$\mathbf{x}_{k+1|k+1} = \mathbf{K}'_{k+1}\mathbf{x}_{k+1|k} + \mathbf{K}_{k+1}\mathbf{y}_{k+1}. \tag{19.37}$$

We next seek the gain matrices K' and K, such that the loss function, as Kalman called it, or given a form that we choose is minimized. Here we are minimizing the conditional mean square analysis error, which is given by

$$\boldsymbol{\varepsilon}_{k+1|k+1}^a = \mathbf{x}_{k+1|k+1} - \mathbf{x}_{k+1}^t. \tag{19.38}$$

A property that we wish for the filter to have is that it is unbiased, which implies that we require $\mathbb{E}[\mathbf{x}_{k+1|k+1}] = \mathbb{E}[\mathbf{x}_{k-1}]$. This is verified through

$$\begin{aligned}\mathbb{E}\left[\mathbf{x}_{k+1|k+1}\right] &= \mathbb{E}\left[\mathbf{K}'_{k+1}\mathbf{x}_{k+1|k} + \mathbf{K}_{k+1}\mathbf{H}\mathbf{x}_{k+1} + \mathbf{K}_{k+1}\boldsymbol{\varepsilon}_{k+1}^o\right], \\ &= \mathbf{K}'_{k+1}\mathbb{E}\left[\mathbf{x}_{k+1|k}\right] + \mathbf{K}_{k+1}\mathbf{H}_{k+1}\mathbb{E}\left[\mathbf{x}_{k+1}\right],\end{aligned} \tag{19.39}$$

where the last term in the first line in (19.39) disappears due to the mean of the observational error being assumed to be zero, i.e., unbiased observational errors.

Before we progress, we return to Kalman's linear dynamic system that describes the evolution of the state, which is defined as

$$\mathbf{x}_{k+1} = \mathbf{M}_k\mathbf{x}_k + \mathbf{B}\mathbf{u}_k. \tag{19.40}$$

Given the dynamical system above, we can then describe the expectation of the state as

$$\begin{aligned}\mathbb{E}\left[\mathbf{x}_{k+1|k}\right] &= \mathbb{E}\left[\mathbf{M}_k\mathbf{x}_{k|k} + \mathbf{B}\mathbf{u}_k\right], \\ &= \mathbf{M}_k\mathbb{E}\left[\mathbf{x}_{k|k}\right] + \mathbf{B}_k\mathbf{u}_k, \\ &= \mathbb{E}\left[\mathbf{x}_{k+1}\right].\end{aligned} \tag{19.41}$$

Hence, if we now combine (19.41) with (19.39), then we have that

$$\mathbb{E}\left[\mathbf{x}_{k+1|k+1}\right] = \left(\mathbf{K}'_{k+1} + \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)\mathbb{E}\left[\mathbf{x}_{k+1}\right], \tag{19.42}$$

and for us to obtain the unbiased condition, we require both sides in (19.42) to be equal. Therefore, we require the factor multiplying the expectation on the right-hand side to be the identity matrix,

$$\begin{aligned} \mathbf{K}'_{k+1} + \mathbf{K}_{k+1}\mathbf{H}_{k+1} &= \mathbf{I}, \\ \Rightarrow \mathbf{K}'_{k} &= \mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}. \end{aligned} \tag{19.43}$$

This implies that the analysis state is given by

$$\begin{aligned} \mathbf{x}_{k+1|k+1} &= \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)\mathbf{x}_{k+1|k} + \mathbf{K}_{k+1}\mathbf{y}_{k+1}, \\ &= \mathbf{x}_{k+1|k} + \mathbf{K}_{k+1}\left(\mathbf{y}_{k+1} - \mathbf{H}_{k+1}\mathbf{x}_{k+1|k}\right), \end{aligned} \tag{19.44}$$

where $\mathbf{K}$ is the **Kalman gain matrix**.

We still have two more step to go. The first step is to derive the analysis error covariance matrix, given the updated analysis state from (19.44), which is denoted as $\mathbf{P}^{\mathbf{a}}_{\mathbf{k+1|k+1}}$ and is defined as

$$\begin{aligned} \mathbf{P}^{a}_{k+1|k+1} &= \mathbb{E}\left[\boldsymbol{\varepsilon}^{a}_{k+1}\left(\boldsymbol{\varepsilon}^{a}_{k+1}\right)^{T}|\boldsymbol{\mathcal{Z}}\right], \\ &= \mathbb{E}\left[\left(\mathbf{x}_{k+1} - \mathbf{x}_{k+1|k+1}\right)\left(\mathbf{x}_{k+1} - \mathbf{x}_{k+1|k+1}\right)^{T}\right], \\ &= \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)\mathbb{E}\left[\boldsymbol{\varepsilon}^{f}_{k}\left(\boldsymbol{\varepsilon}^{f}_{k}\right)^{T}\right]\left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right) + \mathbf{K}_{k+1}\mathbb{E}\left[\boldsymbol{\varepsilon}^{0}_{k+1}\left(\boldsymbol{\varepsilon}^{0}_{k+1}\right)^{T}\right]\mathbf{K}^{T}_{k+1}, \\ &= \left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)\mathbf{P}^{f}_{k+1|k}\left(\mathbf{I} - \mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)^{T} + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}. \end{aligned} \tag{19.45}$$

It is possible to simplify (19.45) as follows:

$$\begin{aligned} \mathbf{P}^{a} &= \mathbb{E}\left[\left(\boldsymbol{\psi}^{a} - \boldsymbol{\psi}^{t}\right)\left(\boldsymbol{\psi}^{a} - \boldsymbol{\psi}^{t}\right)^{T}\right], \\ &= \mathbb{E}\left[\left(\boldsymbol{\psi}^{f} - \boldsymbol{\psi}^{t} + \mathbf{K}\left(\mathbf{y} - \mathbf{y}^{t} - \mathbf{H}\boldsymbol{\psi}^{f} + \mathbf{H}\boldsymbol{\psi}^{t}\right)\right)\left(\boldsymbol{\psi}^{f} - \boldsymbol{\psi}^{t} + \mathbf{K}\left(\mathbf{y} - \mathbf{y}^{t} - \mathbf{H}\boldsymbol{\psi}^{f} + \mathbf{H}\boldsymbol{\psi}^{t}\right)\right)\right], \\ &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbb{E}\left[\left(\boldsymbol{\psi}^{f} - \boldsymbol{\psi}^{t}\right)\left(\boldsymbol{\psi}^{f} - \boldsymbol{\psi}^{t}\right)^{T}\right]\left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)^{T} \mathbf{K}\mathbf{O}\mathbf{K}^{T}, \\ &= \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbf{P}^{f} - \mathbf{P}^{f}\mathbf{H}^{T}\mathbf{K} + \mathbf{K}\left(\mathbf{H}\mathbf{P}^{f}\mathbf{H} + \mathbf{O}\right)\mathbf{K}^{T}. \end{aligned} \tag{19.46}$$

We now have to substitute the expression for the Kalman gain matrix into the left multiplication in the fourth term in (19.46), which cancels the bracketed component of the fourth term, such that only the $\mathbf{P}^{f}\mathbf{H}^{T}$ term remains, which, when combined with $\mathbf{K}^{T}$, cancels with the third term to leave

$$\mathbf{P}^{a} = \left(\mathbf{I} - \mathbf{K}\mathbf{H}\right)\mathbf{P}^{f}. \tag{19.47}$$

The last step that we have to do here is to find the expression for the Kalman gain matrix. We know that we have to minimize the conditional mean square analysis error with respect to the Kalman gain matrix, which is equivalent to

$$L\left(\boldsymbol{\varepsilon}^{a}_{k+1}\right) = \min_{\mathbf{K}_{k+1}} \mathbb{E}\left[\boldsymbol{\varepsilon}^{a}_{k+1}\left(\boldsymbol{\varepsilon}^{a}_{l+1}\right)\middle|\boldsymbol{\mathcal{Z}}_{k+1}\right],$$

Table 19.1 Summary of the Processes Associated With Each Stage of the Kalman Filter and the Definition of the Equivalent Matrix Operations.

Propagation	
Create a forecast, a priori, background, state, $\mathbf{x}_t^{\,f}$	$\mathbf{x}_k^{\,f} = \mathbf{M}_k \mathbf{x}_{k-1}^a + \mathbf{B}_k \mathbf{u}_k$
Propagate the forecast error covariance matrix, $\mathbf{P}_k^{\,f}$	$\mathbf{P}_k^{\,f} = \mathbf{M}_k \mathbf{P}_{k-1}^a \mathbf{M}_k^T + \mathbf{Q}_k$
Update	
Create observation innovation, $\mathbf{d}_k$	$\mathbf{d}_k = \mathbf{y}_k - \mathbf{H}_k \mathbf{x}_k^{\,f}$
Create the Kalman gain matrix, $\mathbf{K}_k$	$\mathbf{K}_k = \mathbf{P}_k^{\,f} \mathbf{H}_k^T \left[\mathbf{H}_k \mathbf{P}_k^{\,f} \mathbf{H}_k^T + \mathbf{R}_k \right]^{-1}$
Update the analysis error covariance matrix, $\mathbf{P}_k^a$	$\mathbf{P}^a = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^{\,f} (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k.$

$$
\begin{aligned}
&= \min_{\mathbf{K}_{k+1}} \text{Trace} \left(\mathbb{E} \left[\boldsymbol{\varepsilon}_{k+1}^a \left(\boldsymbol{\varepsilon}_{l+1}^a \right) \middle| \mathcal{Z}_{k+1} \right] \right), \\
&= \min_{\mathbf{K}_{k+1}} \text{Trace} \left(\mathbf{P}_{k+1|k+1}^a \right). \qquad (19.48)
\end{aligned}
$$

Therefore, differentiating (19.45) with respect to $\mathbf{K}_{k+1}$ and setting to zero yields

$$
\frac{\partial L \left(\boldsymbol{\varepsilon}_{k+1}^a \right)}{\partial \mathbf{K}_{k+1}} = -2 \left(\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1} \right) \mathbf{P}_{k+1|k}^{\,f} \mathbf{H}_{k+1}^T + 2 \mathbf{K}_{k+1} \mathbf{R}_{k+1}^{-1} = \mathbf{0}.
$$

Now, rearranging to isolate $\mathbf{K}_{k+1}$, we obtain the following expression for the Kalman gain matrix:

$$
\mathbf{K}_{k+1} = \mathbf{P}_{k+1|k}^{\,f} \mathbf{H}_{k+1}^T \left[\mathbf{H}_{k+1} \mathbf{P}_{k+1|k}^{\,f} \mathbf{H}_{k+1}^T + \mathbf{R}_{k+1} \right]^{-1}. \qquad (19.49)
$$

Therefore, when dealing with the Kalman filter in the literature, it is quite often broken down into two parts: the first part is known as the **propagation, forecast, or prediction** stage, and the second part is referred to as the **update/analysis** stage. As a way to keep track of the different stages of the Kalman filter, we have provided a summary of the Kalman filter in Table 19.1.

In Chapter 15 we introduced the optimum interpolation formula and mentioned that the gain matrix is similar to that of the Kalman gain matrix in appearance. The main difference we can see now is that the forecast and the analysis error covariance matrices are updated each cycle, rather than having the same isotropic and homogeneous covariance models. This is an advantage of the Kalman filter, as it does contain flow dependency information; however, we were quite reliant on the linear model and observation operator assumptions, as well as on the assumption of Gaussian distribution for the errors.

We shall tackle the first issue here by introducing the **extended Kalman filter**.

19.3 Extended Kalman Filter

The starting point for the extended Kalman filter is to assume that we have a nonlinear discrete-time state-space model of the form

$$
\mathbf{x}_{k+1} = \boldsymbol{\mathcal{M}} \left(\mathbf{x}_k, \mathbf{u}_k, k \right) + \mathbf{w}_k, \qquad (19.50a)
$$

$$
\mathbf{y}_k = \mathbf{h} \left(\mathbf{x}_k, k \right) + \mathbf{v}_k, \qquad (19.50b)
$$

where $\mathbf{w}_k$ and $\mathbf{v}_k$ are Gaussian distributed errors with zero means, and $\boldsymbol{\mathcal{M}}(\mathbf{x}_k, \mathbf{u}_k, k)$ is a nonlinear state transition matrix, while $\mathbf{h}$ is a nonlinear observation operator.

As in the linear case, we shall assume that at time k we have

$$\mathbf{x}_k^a = \mathbb{E}[\mathbf{x}_k | \boldsymbol{\mathcal{Z}}] \quad \text{and} \quad \mathbf{P}_k^a. \tag{19.51}$$

In order to generate the prediction, we need to expand (19.50a) at a Taylor series about the previous cycle's analysis state as

$$\mathbf{x}_{k+1} = \boldsymbol{\mathcal{M}}\left(\mathbf{x}_k^a, \mathbf{u}_k, k\right) + \left(\frac{\partial \boldsymbol{\mathcal{M}}}{\partial \mathbf{x}}\right)\left(\mathbf{x}_k - \mathbf{x}_k^a\right) + \mathcal{O}\left(\left[\mathbf{x}_k - \mathbf{x}_k^a\right]^2\right) + \mathbf{w}_k, \tag{19.52}$$

where the Jacobian/tangent linear model of $\boldsymbol{\mathcal{M}}$ is evaluated at $\mathbf{x}_k^a$. The next step is to take the expectation of (19.52), ignoring the terms above the first order, and assuming that $\mathbf{x}_k^a$ is approximately equal to the conditional mean. This yields

$$\mathbf{x}_k^{\,f} = \mathbb{E}\left[\mathbf{x}_{k+1} | \boldsymbol{\mathcal{Z}}\right] = \boldsymbol{\mathcal{M}}\left(\mathbf{x}_k^a, \mathbf{u}_k, k\right). \tag{19.53}$$

The next step is to develop the forecast error covariance matrix $\mathbf{P}_{k+1}^{\,f}$, but we first have to take into account how the nonlinear state transition matrix affects these calculations. We have to express the forecast error in terms of the tangent linear model, $\mathbf{M}$, which can be done through

$$\begin{aligned}
\boldsymbol{\varepsilon}_k^f &= \mathbf{x}_{k+1} - \mathbf{x}_{k+1}^{\,f}, \\
&= \boldsymbol{\mathcal{M}}\left(\mathbf{x}_k^a, \mathbf{u}_k, k\right) + \mathbf{M}\left(\mathbf{x}_k - \mathbf{x}_k^a\right) + \mathbf{w}_k - \boldsymbol{\mathcal{M}}\left(\mathbf{x}_k^a, \mathbf{u}_k, k\right), \\
&= \mathbf{M}\mathbf{x}_k^a + \mathbf{w}_k.
\end{aligned} \tag{19.54}$$

The forecast error covariance matrix can be found through

$$\begin{aligned}
\mathbf{P}_{k+1}^{\,f} &= \mathbb{E}\left[\mathbf{x}_{k+1}^{\,f}\left(\mathbf{x}_{k+1}^{\,f}\right)^T | \boldsymbol{\mathcal{Z}}\right], \\
&\approx \mathbb{E}\left[\left(\mathbf{M}_k\mathbf{x}_k^a + \mathbf{w}_k\right)\left(\mathbf{M}_k\mathbf{x}_k^a + \mathbf{w}_k\right)^T | \boldsymbol{\mathcal{Z}}\right], \\
&= \mathbf{M}_k\mathbf{P}_k^f\mathbf{M}^T + \mathbf{Q}_k.
\end{aligned} \tag{19.55}$$

We now have to linearize the observation operator, where we use a Taylor series expansion about $\mathbf{x}_{k+1}^{\,f}$, which yields

$$\mathbf{y}_k = \mathbf{h}\left(\mathbf{x}_{k+1}^{\,f}\right) + \mathbf{H}_{k+1}\left(\mathbf{x}_{k+1}^{\,f} - \mathbf{x}_{k+1}\right) + \mathcal{O}\left(\left[\mathbf{x}_{k+1}^{\,f} - \mathbf{x}_{k+1}\right]^2\right) + \mathbf{v}_{k+1}. \tag{19.56}$$

Taking the expectation of (19.56), after it has been truncated to first order, results in

$$\overline{\mathbf{y}}_{k+1} \approx \mathbf{h}\left(\mathbf{x}_{k+1}^{\,f}\right). \tag{19.57}$$

Therefore, the innovation error is

$$\boldsymbol{\varepsilon}_{k+1}^o = \mathbf{y}_{k+1} - \mathbf{h}\left(\mathbf{x}_{k+1}^{\,f}\right), \tag{19.58}$$

which leads to the innovation covariance matrix defined as

$$\begin{aligned}\mathbf{S}_{k+1} &= \mathbb{E}\left[\boldsymbol{\varepsilon}^o_{k+1}\left(\boldsymbol{\varepsilon}^o_{k+1}\right)^T\right],\\ &= \mathbb{E}\left[\left(\mathbf{y}_{k+1}-\mathbf{h}\left(\mathbf{x}^f_{k+1}\right)\right)\left(\mathbf{y}_{k+1}-\mathbf{h}\left(\mathbf{x}^f_{k+1}\right)\right)^T\right],\\ &\approx \mathbb{E}\left[\left(\mathbf{H}_{k+1}\left(\mathbf{x}^f_{k+1}-\mathbf{x}_{k+1}\right)+\mathbf{v}_{k+1}\right)\left(\mathbf{H}_{k+1}\left(\mathbf{x}^f_{k+1}-\mathbf{x}_{k+1}\right)+\mathbf{v}_{k+1}\right)^T\right],\\ &= \mathbf{H}_{k+1}\mathbf{P}^f_{k+1}\mathbf{H}_{k+1}+\mathbf{R}_k. \end{aligned} \tag{19.59}$$

Through following the same argument presented in the last section, it is possible to obtain the same expressions for the Kalman gain matrix, the analysis step, as well as the analysis covariance matrices; the only difference is that they now involve the tangent linear models for the propagation and scaling by the observation operator. Therefore we have

$$\mathbf{K}_{k+1} = \mathbf{P}^f_{k+1}\left(\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)^T \mathbf{S}^{-1}_{k+1}. \tag{19.60}$$

The analysis step is given by

$$\mathbf{x}^a_{k+1} = \mathbf{x}^f_{k+1} + \mathbf{K}_{k+1}\left(\mathbf{y}_{k+1}-\mathbf{h}\left(\mathbf{x}^f_{k+1}\right)\right), \tag{19.61}$$

and the analysis error covariance matrix is given by

$$\mathbf{P}^a_{k+1} = \left(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)\mathbf{P}^f_{k+1}\left(\mathbf{I}-\mathbf{K}_{k+1}\mathbf{H}_{k+1}\right)^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}_{k+1}. \tag{19.62}$$

As we can see, the two approaches of Kalman filter and extended Kalman filter have quite similar structures, but the forecasts and the observation innovations are with respect to the nonlinear operator, and the propagation of the analysis error covariance matrix is through the tangent liner model and its adjoint. We can therefore allow for weakly nonlinear numerical models and observations operators as long as the forecast and observational errors are quite small, so that their prorogation can be described by a linear model between update times. However, there were some numerical stability issues with the original formulation of the Kalman filter, which lead to the **square root Kalman filter**.

19.4 Square Root Kalman Filter

It was soon realized that the original formulations of the Kalman filter were not numerically stable. As the forecast and analysis error covariance matrices are positive definite, they can be represented as $\mathbf{P^f_k} = \mathbf{Z^f_k}\mathbf{Z^{f\,T}_k}$ and $\mathbf{P^a_k} = \mathbf{Z^a_k}\mathbf{Z^{a^T}_k}$, where the matrices $\mathbf{Z^f}$ and $\mathbf{Z^a}$ are the *matrix square root* of the forecast and analysis error covariance matrices, respectively.

Given the square roots of the forecast and analysis error covariance matrices, we are required to change the following two definitions of the original Kalman filter:

$$\mathbf{P^f_k} = \boldsymbol{\mathcal{M}}_k\mathbf{P^a_{k-1}}\boldsymbol{\mathcal{M}}^{\mathbf{t}}_{\mathbf{k}} + \mathbf{Q}, \tag{19.63}$$

$$\mathbf{P^a_k} = \left(\mathbf{I}-\boldsymbol{K}_k\mathbf{H_k}\right)\mathbf{P^f_k}. \tag{19.64}$$

If we assume that there is no model error, then we can write the expression for the forecast error square root covariance matrix above as

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} = \mathcal{M}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k-1}}^{\mathbf{a}}. \tag{19.65}$$

To form the square root of the analysis error covariance matrix is not so straightforward. If we substitute for the Kalman gain matrix into (19.64) and then follow what is referred to as the **Potter method** for the Kalman square root filter update [36], then we have that

$$\begin{aligned} \mathbf{P}_{\mathbf{k}}^{\mathbf{a}} = \mathbf{Z}_{\mathbf{k}}^{\mathbf{a}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{a}^{\mathbf{T}}} &= \left(\mathbf{I} - \mathbf{P}_{\mathbf{k}}^{\mathbf{f}} \mathbf{H}_{\mathbf{k}}^{\mathbf{T}} \left(\mathbf{H}_{\mathbf{k}} \mathbf{P}_{\mathbf{k}}^{\mathbf{f}} \mathbf{H}_{\mathbf{k}}^{\mathbf{T}} + \mathbf{R} \right)^{-1} \mathbf{H}_{\mathbf{k}} \right) \mathbf{P}_{\mathbf{k}}^{\mathbf{f}}, \\ &= \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \left(\mathbf{I} - \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \mathbf{H}_{\mathbf{k}}^{\mathbf{T}} \left(\mathbf{H}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}^{\mathbf{T}}} \mathbf{H}_{\mathbf{k}}^{\mathbf{T}} + \mathbf{R}_{\mathbf{k}} \right)^{-1} \mathbf{H}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \right) \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}^{\mathbf{T}}}, \\ &= \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \left(\mathbf{I} - \mathbf{V}_{\mathbf{k}} \mathbf{D}_{\mathbf{k}}^{-1} \mathbf{V}_{\mathbf{k}}^{\mathbf{T}} \right) \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}^{\mathbf{T}}}, \end{aligned} \tag{19.66}$$

where $\mathbf{V}_{\mathbf{k}} \equiv (\mathbf{H}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}})^{\mathbf{T}}$ and $\mathbf{D}_k \equiv \mathbf{V}_{\mathbf{k}}^{\mathbf{T}} \mathbf{V}_{\mathbf{k}} + \mathbf{R}_{\mathbf{k}}$.

We can then write the square root of the analysis error covariance matrix as

$$\mathbf{Z}_{\mathbf{k}}^{\mathbf{a}} = \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}} \mathbf{X}_{\mathbf{k}} \mathbf{U}_{\mathbf{k}}, \tag{19.67}$$

where we have the property $\mathbf{X}_{\mathbf{k}}$ of $\mathbf{X}_{\mathbf{k}} \mathbf{X}_{\mathbf{k}}^{\mathbf{T}} = (\mathbf{I} - \mathbf{V}_{\mathbf{k}} \mathbf{D}_{\mathbf{k}}^{-1} \mathbf{V}_{\mathbf{k}}^{\mathbf{T}})$, and $\mathbf{U}$ is an arbitrary orthogonal matrix.

We now require a method for finding the square root of the matrix $(\mathbf{I} - \mathbf{V}_{\mathbf{k}} \mathbf{D}_{\mathbf{k}}^{-1} \mathbf{V}_{\mathbf{k}}^{\mathbf{T}})$. As suggested in [439], one way to achieve this goal is to apply a direct approach, where we first solve the linear system

$$(\mathbf{H}_{\mathbf{k}} \mathbf{P}_{\mathbf{k}}^{\mathbf{f}} \mathbf{H}_{\mathbf{k}}^{\mathbf{T}} + \mathbf{R}_{\mathbf{k}}) \mathbf{Y}_{\mathbf{k}} = \mathbf{H}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}}, \tag{19.68}$$

and then form the matrix $\mathbf{I} - \mathbf{V}_{\mathbf{k}} \mathbf{D}_{\mathbf{k}}^{-1} \mathbf{V}_{\mathbf{k}}^{\mathbf{T}} = \mathbf{I} - (\mathbf{H}_{\mathbf{k}} \mathbf{Z}_{\mathbf{k}}^{\mathbf{f}})^{\mathbf{T}} \mathbf{Y}_{\mathbf{k}}$, and its matrix square root $\mathbf{X}_{\mathbf{k}}$ is then computed and applied to $\mathbf{Z}_{\mathbf{k}}^{\mathbf{f}}$ as in (19.67).

As we mentioned in the derivation of the original Kalman filter, it is possible that the time t is before the analysis time and that these approaches are referred to as **smoothers**. We now move on to present briefly a version of the **Kalman smoother**.

19.5 Smoother

There are many different forms of smoothers that are available, and there is a very good summary of some of the different versions of the Kalman smoothers in [73]. In this section we shall present the **Kalman smoother** from [293].

The Kalman smoother according to [293] is a generalization of the Kalman filter which can assimilate future observations as well as past and present observations.

19.5.1 Forward Step: Kalman Filter

The forward part of the smoother consists of the standard Kalman filter, where we have a regular time-independent analysis grid, but where we have an irregular time-dependent observation network. To start

the derivation of the Kalman smoother, we recall the Kalman filter equations:

$$\begin{aligned}
\mathbf{P}_{n+1}^{f} &= \mathbf{M}\mathbf{P}_n^a\mathbf{M}^T + \mathbf{Q}_n, \\
\mathbf{P}_n^a &= (\mathbf{I} - \mathbf{K}_n\mathbf{H}_n)\mathbf{P}_n^f, \\
&= ((\mathbf{P}_n^f)^{-1} + \mathbf{H}_n^T\mathbf{R}_n^{-1}\mathbf{H}_n)^{-1}, \\
\mathbf{K}_n &= \mathbf{P}_n^f\mathbf{H}_n^T(\mathbf{H}_n\mathbf{P}_n^f\mathbf{H}_n^T + \mathbf{R}_n)^{-1}, \\
&= \mathbf{P}_n^a\mathbf{H}^T\mathbf{R}_n^{-1}, \\
\mathbf{x}_{n+1}^{f} &= \mathbf{M}_n\mathbf{x}_n^a, \\
\mathbf{x}_n^a - \mathbf{x}_n^f &= \mathbf{K}_n(\mathbf{y}_n - \mathbf{H}_n\mathbf{x}_n^f).
\end{aligned}$$

To go backward, we now consider the **reverse-time information filter**.

19.5.2 Backward Step: Reverse-Time Information Filter

According to [293], the reverse-time information filter can be thought of as a Kalman filter that runs backward in time. This new filter is derived in [84], but we shall summarize the main steps and arguments here.

We start with the fact that the true state vector on the analysis grid at time t_n can be obtained from the corresponding vector at t_{n-1} through the equation

$$\mathbf{x}_n = \mathbf{M}_{n-1}\mathbf{x}_{n-1} + \mathbf{e}_{n-1}^q, \tag{19.69}$$

where $\mathbf{e}_n^q$ is the model error growth between t_{n-1} and t_n, and $\mathbf{x}_n$ is the true state at t_n. If the model was able to run backward in time, then we could define $\mathbf{M}_n^{-1}$ such that

$$\mathbf{x}_{n-1} = \mathbf{M}_{n-1}^{-1}(\mathbf{x}_n - \mathbf{e}_{n-1}^q). \tag{19.70}$$

Let us suppose that we have a future analyzed state and we integrate the model backward as shown in (19.70). In [293] the authors introduce the superscripts β to represent a **hindcast** and α to indicate an analysis that is produced from present and future data only. Therefore we have

$$\mathbf{x}_{n-1}^{\beta} = \mathbf{M}_{n-1}^{-1}\mathbf{x}_n^{\alpha}. \tag{19.71}$$

Subtracting (19.70) from (19.69) results in

$$\mathbf{e}_{n-1}^{\beta} = \mathbf{M}_{n-1}^{-1}(\mathbf{e}_n^{\alpha} + \mathbf{e}_{n-1}^q), \tag{19.72}$$

where $\mathbf{e}_{n-1}^{\beta} = \mathbf{x}_n^{\beta} - \mathbf{x}_n$ and $\mathbf{e}_n^{\alpha} = \mathbf{x}_n^{\alpha} - \mathbf{x}_n$ are the hindcast and analysis errors, respectively.

If we now right-multiply (19.72) by $(\mathbf{e}_{n-1}^{\beta})^T$ and apply the expectation operator, then we obtain

$$\mathbf{P}_{n-1}^{\beta} = \mathbf{M}_{n-1}^{-1}(\mathbf{P}_n^{\alpha} + \mathbf{Q}_{n-1})(\mathbf{M}_{n-1}^{-1})^T, \tag{19.73}$$

where $\mathbf{P}_n^{\beta} = \mathbb{E}[\mathbf{e}_n^{\beta}(\mathbf{e}_n^{\beta})^T]$ and $\mathbf{P}_n^{\alpha} = \mathbb{E}[\mathbf{e}_n^{\alpha}(\mathbf{e}_n^{\alpha})^T]$ are the **hindcast and reverse time analysis error covariance matrices**, respectively.

The next step in the derivation of the reverse-time information filter is to find a way to rewrite (19.71) and (19.73) without the inverse matrices. The trick is to write the inverse of the model in the form of a transpose instead. This is achieved through the property that for any error covariance matrix, $\mathbf{P}$, then we can define a corresponding information matrix $\mathbf{N} = \mathbf{P}^{-1}$. Next we introduce a new state variable $\mathbf{z} = \mathbf{N}\mathbf{x} = \mathbf{P}^{-1}\mathbf{x}$.

If we now substitute $\mathbf{N}$ as just defined into (19.71) and (19.73), then we end up with

$$\begin{aligned}\mathbf{z}_{n-1}^{\beta} &= \mathbf{N}_{n-1}^{\beta}\mathbf{M}_{n-1}^{-1}(\mathbf{N}_n^{\alpha})^{-1}\mathbf{z}_n^{\alpha},\\ &= \mathbf{M}_{n-1}^{T}(\mathbf{I}+\mathbf{N}_n^{\alpha}\mathbf{Q}_{n-1})^{-1}\mathbf{z}_n^{\alpha},\\ \mathbf{N}_{n-1}^{\beta} &= \mathbf{M}_{n-1}^{T}(\mathbf{I}+\mathbf{N}_n^{\alpha}\mathbf{Q}_{n-1})^{-1}\mathbf{N}_n^{\alpha}\mathbf{M}_{n-1}.\end{aligned} \tag{19.74}$$

However, we still require definitions for $\mathbf{N}_n^{\alpha}$ and $\mathbf{z}_n^{\alpha}$, which are defined as

$$\mathbf{N}_n^{\alpha} = \mathbf{N}_n^{\beta} + \mathbf{H}_n^{T}\mathbf{R}_n^{-1}\mathbf{H}_n, \tag{19.75}$$

$$\mathbf{y}_n^{\alpha} = \mathbf{y}_n^{\beta} + \mathbf{H}_n^{T}\mathbf{R}_n^{-1}\mathbf{z}_n. \tag{19.76}$$

Thus the backward filter is started at some time t_N, through setting $\mathbf{z}_N^{\beta}$ and $\mathbf{N}_N^{\beta}$ to zero; this implies that there is no information for $t > t_N$. In [293] the authors comment that the analysis step in the reverse-time information filter is quite simple (19.76) compared to the equivalent step in the Kalman filter, while the hindcast step (19.74) is more complex.

Now, given our forward and backward estimates of the state vector, we have to define a way to combine them.

19.5.3 Smoothing

At any time t_n, then we have two estimates for the state vector: one obtained from the past information using the Kalman filter and one obtained from the future information using the reverse-time information filter. The **smoothing** step combines these two estimates optimally in a minimum variance estimate sense.

It is highlighted in [293] that there are three possible combination that could be used:

- Combine $\mathbf{x}_n^{a}$, which uses observations t_n, with $\mathbf{z}_n^{\beta}$ that does not.
- Combine $\mathbf{x}^{f}$ with $\mathbf{z}_n^{\alpha}$.
- Combine $\mathbf{x}^{f}$ with $\mathbf{z}_n^{\beta}$ and $\mathbf{y}_n$.

An important feature to note about the combinations above is that the observations are only ever used once.

In this last stage of the smoother, we shall denote the smoothed state at time t_n as $\mathbf{x}_n^{s}$, and the corresponding smoother analysis error covariance matrix by $\mathbf{P}_n^{s}$. The optimal combination is derived in [85], but it can be shown that these estimates are given by

$$\begin{aligned}(\mathbf{P}_n^{s})^{-1} &= (\mathbf{P}_n^{a})^{-1} + \mathbf{N}_n^{\beta},\\ \Rightarrow \mathbf{P}_n^{s} &= (\mathbf{I}+\mathbf{P}_n^{a}\mathbf{N}_n^{\beta})^{-1}\mathbf{P}_n^{a},\end{aligned} \tag{19.77}$$

$$\mathbf{x}_n^{s} = (\mathbf{I}+\mathbf{P}_n^{a}\mathbf{N}_n^{\beta})^{-1}(\mathbf{x}_n^{a}+\mathbf{P}_n^{a}\mathbf{z}_n^{\beta}). \tag{19.78}$$

Therefore, in summary the Kalman smoother equations are the Kalman filter's equations, the reverse-time information filter equations (19.74) and (19.76), and finally the smoothing equation (19.77).

19.6 Properties and Equivalencies of the Kalman Filter and Smoother

In this section we summarize some properties and linkages to other data assimilation systems.

Direct insertion

While at first it may not appear obvious how the Kalman filter equations could be related to the nudging techniques, it is shown in [85], that the direct insertion techniques simply take the forecast update step and directly replace the elements of $\mathbf{x}_n^a$ with spatially interpolated observations.

Optimal/statistical interpolation

We made a passing reference to this equivalency earlier; to recap, the OI approach uses the forecast step, the analysis step, and the Kalman gain matrix update. However, must recall that the OI schemes are based on localized grid points/volumes, whereas the Kalman filter made no restrictions; as in 3D VAR, the Kalman filter is a global solution. Also, apart from having to describe the initial analysis covariance matrix and the model error covariance matrix, the flow evolves the analysis forecast error covariance matrices, where as for the OI, the forecast error covariance matrix is a prescribed set of functions to ensure certain decorrelation length scales as well as to maintain balances.

Kalman filter equivalent to 4D VAR

It was shown in [259] that over the same time interval $[0, N]$, and assuming a perfect model, i.e., $\mathbf{Q} \equiv 0$, and assuming that both algorithms use the same data, where in particular $\mathbf{P}_0^{\,f} = \mathbf{0}$, then there is equality of:

1. the final analysis $\mathbf{x}_N^a$ produced by the Kalman filter algorithm; and
2. the final value of the optimal trajectory estimated by the 4D VAR.

That is to say $\mathbf{x}_N^a \equiv \mathbf{M}_{0,N}\mathbf{x}^a(0)$.

Balance

As mentioned above, the Kalman filter, as derived, is not designed to prevent the initial conditions from exciting spurious waves in the dynamical systems [85]. However, it is possible to introduce frequency-selective damping to reduce the influence of these spurious modes. Another technique is to consider the modified Kalman filter where the Kalman gain matrix is multiplied by a slow manifold projection matrix. **Note:** The resulting gain matrix could be suboptimal.

Equivalent to observer-based feedback

Recalling the feedback theory from the control theory chapter, then we have that, given the discrete time control system, the optimal observer for the completely observable time invariant system

$$\mathbf{x}_{k+1} = \mathbf{M}\mathbf{x}_k + \mathbf{B}\mathbf{u} + \mathbf{w}_k,$$
$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k,$$

is given by

$$\mathbf{x}_{k+1}^{a} = (\mathbf{I} - \mathbf{KH})\left(\mathbf{M}\mathbf{x}_{k}^{a} + \mathbf{B}\mathbf{u}_{k}\right) + \mathbf{K}\mathbf{y}_{k+1},$$
$$\mathbf{K} = \mathbf{PH}\left(\mathbf{HPH}^{T} + \mathbf{R}\right)^{-1},$$

where $\mathbf{P}$ is the positive definite solution of

$$\mathbf{P} = \mathbf{M}(\mathbf{P} - \mathbf{PH}^{T}(\mathbf{R} + \mathbf{HPH}^{T})^{-1})\mathbf{M} + \mathbf{Q}. \tag{19.79}$$

The matrix equation in (19.79) is the algebraic Riccati equation that we saw in the Chapter 7.

19.7 Summary

In this chapter we have introduced the Kalman filter from two different approaches; the first approach was through following the original 1960 paper, and we must recall that in that paper, Kalman does not have any observation error, so there is no $\mathbf{R}_n$ matrix in the expression for the innovation error covariance matrix, as it is referred to nowadays. As a reminder, the measurement error does not appear until [206]. The second approach for the derivation of the filter was from an unbiased estimator formulation, where this time we did include observational error.

The next part of this chapter introduced the extended Kalman filter, where we allow for nonlinear numerical models and observation operators, but we apply a Taylor series expansions to both of the nonlinear components to be able to apply the updates and the evolution of the forecast and analysis error covariance matrices. However, the updates to the analysis state included the nonlinear observation operator, and the forecast state comes from running the nonlinear model forward.

After the extended Kalman filter we introduced what [293] refers to as the **Kalman smoother**, but we acknowledge that there are several other smoothers that are based upon Kalman theory. One such smoother is the fixed-lag smoother [70].

We ended this chapter with some properties, equivalencies to other data assimilation schemes.

The Kalman filter is quite an attractive data assimilation scheme because of the flow dependencies that it carries forward in the propagation of the analysis and forecast error covariance matrices, albeit linearly. A drawback of the Kalman filter is that it is heavily dependent on the Gaussian assumption, whereas the variational approaches are not [129,132,135–137]; see Chapter 21.

Before we leave the Kalman filter, we add a little history to the use of the filter. It was not very widely known until quite recently that the Kalman filter, or rather the square root version of the filter, played an important part in the Apollo missions to the moon [286].

However, there has been much work in the hydrology, ocean, and atmospheric prediction communities, to name but a few, to find a way to relax the linear assumption of the Kalman filter, or to be able to use the analysis state, which is the mean of a Gaussian distribution, to bring in flow dependence into the covariance matrix. These approaches are referred to as **ensemble Kalman filters** and we shall now introduce the broader topic of **ensemble-based data assimilation**.

This page intentionally left blank

CHAPTER

Ensemble-Based Data Assimilation

20

Contents

The first application of what would become the **ensemble Kalman filter**, more commonly known as the **EnKF**, appears in Evensen's pioneering 1994 paper titled: *Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics* [111]. We warn you now that there are going to be a lot of acronyms in this chapters! The motivation for the introduction of the EnKF comes from work by Evensen in 1992 and 1993, where he was applying the extended Kalman filter, **EKF**, to a multilayered quasi-geostrophic ocean model [109,110]. It was discovered in [109,110] that there was a closure problem in the error covariance evolution equation. It is stated in [111] that the EKF applies a closure scheme where the third- and higher-order moments in the error covariance evolution equation are discarded. The simple closure technique applied in the EKF results in an unbounded error variance growth caused by the linearization that is performed.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00024-3

The important conclusion from [109,110] was that a sequential data assimilation algorithm provides good results in a data assimilation scheme for the nonlinear quasi-geostrophic model, and that the results improved significantly according to improvements in the error estimate for the model forecast.

This was the motivation to develop the EnKF by Evensen, which has now evolved to have many different versions, of both the Kalman filter, but also the smoother, and even to optimum interpolation (OI). There are now versions of the EnKF in operational use for different geophysical situations.

20.1 Stochastic Dynamical Modeling

Before we present the derivation of the EnKF, we briefly present some of the statistical/stochastic theory that Evensen uses to justify the EnKF. Following the flow of the 1994 paper, the first stochastic process we present is **Liouville's theorem**, which leads to Liouville's equations:

Liouville's theorem and equation

Theorem 20.1. *Consider a dynamical system with coordinates q_i and conjugate momenta p_i, where $i = 1, 2, \ldots, N$. Then the phase space distribution $\rho(p, q)$ determines the probability $\rho(p, q)\, d^N q d^N p$ that a particle will be found in the infinitesimal phase space volume $d^N d^N$. The associated Liouville equation governs the evolution of the density in time, where now $\rho(p, q; t)$ is a function of time and the equation of the evolution is*

$$\frac{d\rho}{dt} = \frac{\partial \rho}{\partial t} + \sum_{i=1}^{N} \left(\frac{\partial \rho}{\partial q_i} \dot{q}_i + \frac{\partial \rho}{\partial p_i} \dot{p}_i \right) = 0. \tag{20.1}$$

We now introduce the notion of **Brownian motion**, which is defined as follows.

Definition 20.2. A linear Brownian motion, $b(t)$, is a real-valued stochastic process $\{b(t) : t \geq 0\}$ with the following properties:

1. $b(0) = x$.
2. The process has **independent increments**, which is to say that for all times $0 \leq t_1 \leq t_2 \leq \cdots \leq t_n$, then the increments $b(t_n) - b(t_{n-1}), b(t_{n-1} - t_{n-2}), \ldots, b(t_2) - b(t_1)$ are independent variables.
3. For all $t \geq 0$ and δt, the increment $b(t + \delta t) - b(t)$ are Gaussian distributed with expectation zero and variance δt.
4. The function $t \mapsto b(t)$ is continuous.

The multivariate version of Brownian motion is referred to as the n-dimensional Brownian motion and is defined as follows.

Definition 20.3. If $b_1, b_2, \ldots, b_n$ are independent linear Brownian motions stated in $x_1, x_2, \ldots, x_n$, then the stochastic process $\{\boldsymbol{b}(t) : t \geq 0\}$, where

$$\{\boldsymbol{b}\} = \{b_1(t), b_2(t), \ldots, b_n(t)\}, \tag{20.2}$$

is called an n-dimensional Brownian motion.

We now introduce the definition of the Markov property, which leads to the definition of the Markov process as follows.

Definition 20.4. A stochastic process is said to have the **Markov property** if the conditional probability distribution of the future states of the process, which could be dependent on both past and present states, depends only upon the present state.

Definition 20.5. A **Markov process** is a stochastic model that has the Markov property. It can be used to model a random system that changes states according to a transition rule that only depends on the current state. For a discrete time situation, this process is referred to as a **Markov chain**.

20.2 Ensemble Kalman Filter

As mentioned earlier in this chapter, the EnKF was first introduced in the ground-breaking paper by Geir Evensen in 1994 [111]. An important statement made at the start of the theory of stochastic dynamic prediction section of [111] is still relevant today:

> The choice of another interpolation scheme of just different statistical parameters in the interpolation scheme will produce another initial state resulting in a different forecast even if the same deterministic model is applied. It is not possible to say that the forecast based on any interpolated initial conditions is right or wrong or better or worse, since each initial state estimate represents an individual member of an infinite ensemble of possible states that are consistent with the data.

Evensen then quotes a very telling statement from a paper by Epstein in 1969 [104]:

> The different analyses will yield different forecasts, even if each were submitted to the same forecast procedure. If there is no way of determining which, if any, analysis is right, and since none is known to be wrong, there is no way of knowing in any instance, which to believe.

Given these motivations, we move on to derive the EnKF. We start by denoting the state of any geophysical system as $\boldsymbol{\psi}(t) \in \mathbb{R}^n$, where the component in the state vector are the vales of all dependent variables in the model of consideration. The state vector at a specified time $\boldsymbol{\psi}(t)$ can be represented by a single point in an n-dimensional phase space, $\mathcal{D}$. Therefore, the time evolution of the state vector $\boldsymbol{\psi}(t)$ is described by the continuous motion of the point along a trajectory in phase space.

The uncertainty in the initial state vector can be represented by a large ensemble of possible initial states, each assigned an individual probability value. We have a copy of the schematic of a cloud of phase points surrounding the analyzed initial estimate $\boldsymbol{\psi}(t=0)$ in Fig. 20.1 from [111]. We now suppose that there are N points, where we assume that N is quite large, and dN is their density, which is defined as points per volume increment, at any location. As $N \to \infty$, as such it is possible to define a probability density function of a distribution given by

$$\phi(\psi) = \frac{dN}{N}, \tag{20.3}$$

which can vary throughout the space. We now define the expression

$$\phi(\psi)\, d\psi, \tag{20.4}$$

as the probability of a phase of the system being located at a certain instant inside the n-dimensional volume element $d\boldsymbol{\psi}$ located around the point ψ. The probability density, ϕ, must be defined over all

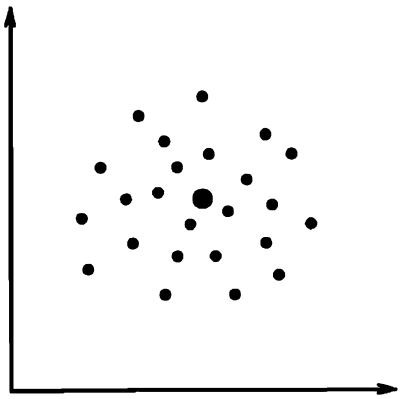


FIGURE 20.1

Copy of Evensen's ensemble schematic from [111].

of the phase space, and $\phi \geq 0$ for all ψ and t. Given a probability density function, $\phi(\boldsymbol{\psi})$, then the expected value of a quantity of $h(\boldsymbol{\psi})$ is defined as

$$\mathbb{E}[h(\psi)] = \int_a^b h(\boldsymbol{\psi})\,\phi(\boldsymbol{\psi})\,d\boldsymbol{\psi}. \tag{20.5}$$

If we recall the definition for the centered moments, remembering that these are not the actual moments of the probability density, then we have

$$\boldsymbol{\mu}_i = \mathbb{E}\left[\boldsymbol{\psi}_i\right], \tag{20.6a}$$
$$\mathbf{P}_{ij} = \mathbb{E}\left[\left(\boldsymbol{\psi}_i - \boldsymbol{\mu}_i\right)\left(\boldsymbol{\psi}_j - \boldsymbol{\mu}_j\right)\right], \tag{20.6b}$$
$$\boldsymbol{\Theta}_{ijk} = \mathbb{E}\left[\left(\boldsymbol{\psi}_i - \boldsymbol{\mu}_i\right)\left(\boldsymbol{\psi}_j - \boldsymbol{\mu}_j\right)\left(\boldsymbol{\psi}_k - \boldsymbol{\mu}_k\right)\right], \tag{20.6c}$$
$$\boldsymbol{\Gamma}_{ijkl} = \mathbb{E}\left[\left(\boldsymbol{\psi}_i - \boldsymbol{\mu}_i\right)\left(\boldsymbol{\psi}_j - \boldsymbol{\mu}_j\right)\left(\boldsymbol{\psi}_k - \boldsymbol{\mu}_k\right)\left(\boldsymbol{\psi}_i - \boldsymbol{\mu}_i\right)\right]. \tag{20.6d}$$

An important feature to note here is that Evensen refers to the expressions above as the moments of the distribution. That is not the correct description for them. They are in fact part of the comoments. Recalling the expressions for cokurtosis and coskewness, we see that the expressions in (20.6c) and (20.6d) are the definitions for one of the numerator of the coskewness and cokurtosis. If $k, l, j = i$ then we have the corrected moment, but again only the numerator of skewness and kurtosis as we are not dividing by the required power of the variance.

It is usually assumed that the PDF for the initial state, which is denoted by $\phi(\psi(t=0))$, is a multivariate Gaussian distribution with a specified vector of means $\boldsymbol{\mu}(t=0)$ and a covariance matrix $\mathbf{P}(t=0)$. An important feature mentioned in [111], and should be noted here, is that at the initial time, the estimate for the mean is the initial state of the field, i.e., $\boldsymbol{\psi}(t=0) = \boldsymbol{\mu}(t=0)$. The covariance matrix at the initial time represents the uncertainty of the initial state, and the probability decreases when you move away from the mean at $t=0$.

In the way Evensen has set up this problem, the time evolution of the different geophysical states is described by continuous motion of the respective points through the phase space, and is a determin-

istic process in which a given initial state generates a definite path into the future. We have a copy of Evensen's schematic from [111] in Fig. 20.2, representing the time evolution of an ensemble of phase points.

The statement that Evensen makes next is not strictly true; Evensen states that during the time integration the central forecast, which is the analyzed state from a previous assimilation step, may drift away from the forecasted mean state, now here is where Evensen needs a disclaimer, as he refers to this state as the **most probable state**. The forecast mean state will only be the most probable state if the PDFs are Gaussian, or if the PDF is symmetric. It is in fact the forecasted minimum variance state.

As we can see from Fig. 20.2, the ensemble forecast may deform and expand or contract in size, where the predicted errors are determined by the variance of the forecasted ensemble [111].

If we now consider the time evolution of the PDF containing all the statistical information about the ensemble, then this ensemble of possible states moves through the phase space that is governed by the dynamic laws of the geophysical situation begin modeled. The density of the ensemble also has an associated probability, so that the dynamic equations can be used to generate probability predictions, as well as predictions of the physical state itself.

Each of the initial states in a volume element, $d\boldsymbol{\psi}$, will evolve according to some deterministic model, and no members of the phase points in $\boldsymbol{\psi}$ may be created or lost, which is equivalent to the **conservation law of probability** which is stated as

$$\frac{\partial \phi}{\partial t} + \sum_{i=1}^{n} \frac{\partial \dot{\psi} \phi}{\partial \psi_i} = 0, \tag{20.7}$$

where $\dot{\psi}_i = \frac{\partial \psi_i}{\partial t}$ is referred to as the **velocity** component in the ψ_i direction, which is determined by the dynamical system we are considering. Equation (20.7) states that the local change of probability density with time in phase space $\frac{\partial \phi}{\partial t}$ balances the probability flux represented by the remaining terms in the summation. An important feature that is noted in [111] at this point is that given appropriate boundary conditions for ϕ, for example, $\phi \to 0$ as $\psi_i \to \pm\infty$, and an initial probability density $\phi(\psi, t = 0)$,

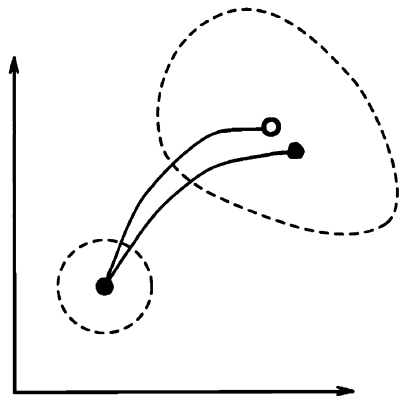


FIGURE 20.2

Copy of Evensen's evolved ensemble PDF from [111].

then (20.7) can be integrated to obtain $\phi(\psi, t)$ for $t \geq 0$. However, the integration by direct numerical methods is impractical because of the size of the problem.

The continuity equation for the PDF ϕ provides a criterion for error growth/decay, and can be written as

$$\frac{\partial \phi}{\partial t} + \sum_{i=1}^{n} \psi_1 \frac{\partial \phi}{\partial \phi_i} = -\phi \sum_{i=1}^{n} \frac{\partial \dot{\psi}_i}{\partial \psi}, \tag{20.8}$$

according to [111]. The right-hand side of (20.8) is a divergence term that describes either the increase or decrease in the PDF, or it can be seen as the contraction or expansion of the cloud of phase points. When the right-hand side of (20.8) is equal to zero, then (20.8) reduces to the Liouville equation presented earlier. This implies that the density of the ensemble is conserved with the motion along the path in phase space.

We now consider the volume element $d\mathcal{D}$ in phase space at time t, containing a cloud of points representing states with specific probabilities. At time $t + \delta t$, the points from $d\mathcal{D}$ fill another volume $d\mathcal{D}'$ of the phase space. As we saw in the last section, Liouville's theorem states that $d\mathcal{D} = d\mathcal{D}'$, which implies that no points are lost or gained from the volume during the integration; the probability of finding the system in $d\mathcal{D}'$ is equal to the probability of finding it in $d\mathcal{D}$.

An important note that Evensen makes at this point is that the volume is still freely deformable, diverging in one dimension and converging in another, and the variance of the ensemble may change at different positions even though the energy for each of the ensemble members is conserved.

If we now consider a general geophysical dynamical model, then in the theory of data assimilation we assume that this system contains errors. These errors are assumed to be random white processes with a specified variance; mathematically this is equivalent to

$$d\psi = \mathbf{g}(\boldsymbol{\psi}, t) + d\boldsymbol{\varepsilon}, \tag{20.9}$$

where $d\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is a vector of white noise with a mean equal to zero. Equation (20.9) is an **Itô stochastic differential equation** that describes a Markov process. The evolution of the probability density function for this equation is described by the forward **Kolmogorov's** equation, which is also referred to as the **Fokker-Planck** equation, given by

$$\frac{\partial \phi}{\partial t} + \sum_{i=1}^{n} \frac{\partial \mathbf{g}_i \phi}{\partial \psi_i} = \sum_{i,j=1}^{n} \frac{\mathbf{Q}_{i,j}}{2} \frac{\partial^2 \phi}{\partial \psi_i \partial \psi_j}, \tag{20.10}$$

where $\mathbf{Q}$ is again the model error covariance matrix, defined as $\mathbf{Q} = \mathbb{E}\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^T\right]$. The proof of equation (20.10) can be found in [198].

Evensen goes on to derive a Taylor series expansion about the mean for a quasi-geostrophic model. This leads to the expression for the evolution of the mean, and the covariance matrix in terms of the higher-order moments, as well as the model evolution operators. We shall not go into detail about these equations but the reader is referred to [111] for more information. However, the expansions for the evolution of the mean and covariance motivate a discussion about Monte Carlo method approximations. Now Evensen does not introduce the equations for the EnKF in [111]; they are only described by text, but the equations using the ensemble do appear in [53]. However, it is the introduction and discussion about the EnKF that we summarize here.

After the results from the experiments with the Monte Carlo scheme with the evolution of the mean and the covariance matrix, Evensen indicates that if this approach is to be considered for data assimilation, then there are still two important components of the analysis scheme that need to be addressed: (1) assimilating the measurement and (2) calculating updates to the state estimates. Taking these two points in turn, we have the following.

An important feature to notice about (20.10) that Evensen highlights is that the stochastic forcing introduces a diffusion term that tends to flatten the probability, which is equivalent to spreading the ensemble, during the integration, which is to say that the probability decreases, and as such the errors increase.

If it is were easy or possible to solve (20.10), then we would calculate the mean and the covariance of the errors at different times. For a nonlinear model as we would be using in most geophysical situations, Evensen points out that the mean and the covariance matrix will not in general characterize $\phi(\boldsymbol{\psi}, t)$. However, the first two moments do determine the mean path and the dispersion about that path.

Therefore, given these complicated expression, Evensen turns to Monte Carlo methods where he states that a large cloud of state, that are points in phase space, can be used to represent a specific PDF. Thus, given this approximation to the PDF, we could integrate the ensemble of states forward in time, where it would be easy to calculate approximate estimates for the moments of PDF at different time levels. It is claimed in [111] that when the ensemble size N increases, the errors in the solution for the PDF will approach zero at a rate of $\frac{1}{\sqrt{N}}$, and when we have ensembles of the order of hundreds, then the error will be dominated by statistical noise rather than by dynamical errors.

We now leave [111] as the equations for the EnKF are not presented here, but a written explanation of the procedure is given. The equations for the EnKF first appear in Evensen and Van Leeuwen's paper, *Assimilation of Geosat altimeter data for the Agulhas current using the ensemble Kalman filter with a quasi-geostrophic model*, in 1996 [112]. We start by introducing the matrix $\mathbf{A}$ of dimensions $n \times N$, where n is the number of state variables and N is the number of ensemble members, where the geophysical model's state from each ensemble member is stored. If we now denote the ensemble forecasts as $\mathbf{A}_k^f$ for time step k, then we can calculate the ensemble forecast error covariance matrix at time k, denoted by $\mathbf{P}_k^f$, as

$$\mathbf{P}_k^f \equiv \frac{1}{N-1}\left(\mathbf{A}_k^f - \overline{\mathbf{M}}_k^f\right)\left(\mathbf{A}_k^f - \overline{\mathbf{M}}_k^f\right)^T, \tag{20.11}$$

where $\overline{\mathbf{M}}_k^f$ is an array that contains the predicted ensemble mean in each column. We should note that the rank of the error covariance matrix $\mathbf{P}_k^f$ will be less than or equal to the number of members in the ensemble. We briefly recall the analysis equations in the Kalman filter that the EnKF is going to approximate:

$$\boldsymbol{\psi}_k^a = \boldsymbol{\psi}_k^f + \mathbf{K}\left(\mathbf{y}_k - \mathbf{H}_k\boldsymbol{\psi}_k^f\right), \tag{20.12}$$

$$\mathbf{y}_k = \mathbf{H}_k\boldsymbol{\psi}_k^f + \boldsymbol{\varepsilon}_k^o, \tag{20.13}$$

$$\mathbf{K}_k = \mathbf{P}_k^f\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_k^f\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1}. \tag{20.14}$$

We now consider the Kalman gain matrix in (20.14), without the observation error, where this part of the matrix, referred to as the **representer matrix** in [112], given by $\widehat{\mathbf{R}}_k \equiv \mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T$, will be the least of the rank of $\mathbf{P}_k^f$, and the rank of $\mathbf{H}_k$.

We must add a note of caution here about the reduced rank property. If the number of observations is greater than the number of ensemble members, then the representer matrix will be singular. In [112] the authors suggest a method to overcome the rank problem. They also highlight that the covariance functions calculated from the ensemble should still give a good estimate of the true covariance functions, and of course if we can increase the number of ensemble members, then this approximation will aid in reducing the statistical noise from the undersampling.

We now rewrite (20.12), the analysis equation, as

$$\boldsymbol{\psi}_k^a = \boldsymbol{\psi}_k^f + \mathbf{B}_k^T \mathbf{b}_k, \tag{20.15}$$

where $\mathbf{B}_k \equiv \mathbf{H}_k \mathbf{P}_k^f$, and the rows in $\mathbf{B}_k$ are influence functions, but we have seen them earlier as part of the PSAS system as the representers for each observation, then the vector $\mathbf{b}_k$ contains the amplitudes for each influence functions and is found through solving the matrix-vector system

$$\left(\mathbf{H}_k \mathbf{P}_k^f \mathbf{H}_k^T + \mathbf{R}_k\right) \mathbf{b}_k = \mathbf{y} - \mathbf{H}_k \boldsymbol{\psi}_k^f. \tag{20.16}$$

We should note that it is not necessary to calculate the full error covariance matrices in (20.15) and (20.16), but we can apply the following procedure to evaluate them:

1. Calculate $\mathbf{S}_k = \frac{1}{N}\mathbf{H}_k \left(\mathbf{A}_k^f - \overline{\mathbf{M}}\right)$, which is a $N_o \times N$ matrix, where N_o represents the number of observations.
2. Form the influence functions matrix $\mathbf{B}_k = \mathbf{S}_k \left(\mathbf{A}_k^f - \overline{\mathbf{M}}_k^f\right)^T (N-1)^{-1}$.
3. Form the representer matrix $\widehat{\mathbf{R}}_k = \mathbf{H}_k \mathbf{P}_k^f \mathbf{H}_k^T \equiv \mathbf{H}_k \mathbf{B}_k^T \equiv \mathbf{S}_k \mathbf{S}_k^T (N-1)^{-1}$.

Therefore to find the coefficient $\mathbf{b}_k$, we simply solve (20.16), but only have to calculate $\mathbf{S}_k$, as long as $\widehat{\mathbf{R}}_k + \mathbf{R}_k$ is non-singular.

Returning briefly to [111], it is shown here that an ensemble with the correct analyzed error statistics must be calculated during each analysis. This can be achieved if the same gain matrix $\mathbf{K}_k$ is used in the analysis step in (20.15) for each ensemble member, where each member provides a different $\boldsymbol{\psi}_l^f$, for $l = 1, 2, \ldots, n$, then they change the right-hand side of (20.16).

However, to find the EnKF analysis error covariance matrix equation, we consider [52], where we are reminded that the true forecast and analysis covariances from the Kalman filter are given by

$$\mathbf{P}_k^f = \mathbb{E}\left[\left(\boldsymbol{\psi}_k^f - \boldsymbol{\psi}_k^t\right)\left(\boldsymbol{\psi}_k^f - \boldsymbol{\psi}_k^t\right)^T\right], \tag{20.17a}$$

$$\mathbf{P}_k^a = \mathbb{E}\left[\left(\boldsymbol{\psi}_k^a - \overline{\boldsymbol{\psi}}_k\right)\left(\boldsymbol{\psi}_k^a - \overline{\boldsymbol{\psi}}_k\right)^T\right]. \tag{20.17b}$$

In the EnKF we do not know the true covariances; this is also true for **all** forms of data assimilation, but in the EnKF the true state is replaced by the **ensemble covariance matrices** that are calculated around the ensemble mean $\overline{\boldsymbol{\psi}}$. Therefore, it is possible to approximate (20.17a) and (20.17b) by

$$\mathbf{P}_{e,k}^f = \mathbb{E}\left[\left(\boldsymbol{\psi}_k^f - \boldsymbol{\psi}_k^t\right)\left(\boldsymbol{\psi}_k^f - \boldsymbol{\psi}_k^t\right)^T\right], \tag{20.18a}$$

$$\mathbf{P}_{e,k}^a = \mathbb{E}\left[\left(\boldsymbol{\psi}_k^a - \boldsymbol{\psi}_k^t\right)\left(\boldsymbol{\psi}_k^a - \boldsymbol{\psi}_k^t\right)^T\right]. \tag{20.18b}$$

In summary; the EnKF is an approach that is equivalent to solving the Fokker-Planck (Kolmogorov's) equation for the evolution of the PDF for the error statistics [449].

We now move on to different formulations of the EnKF; where the first of these is the **perturbed observations**-based EnKF from [52].

20.2.1 Perturbed Observations-Based EnKF

The starting point of the perturbed observations-based EnKF is the ensemble estimates for the forecast and analysis error covariance matrices from (20.17a) and (20.17b), where the argument for perturbing the observations comes from the assumption that there is no representative error involved here. If we recall that the analysis error covariance matrix from the Kalman filter is defined as

$$\mathbf{P}^a = \left(\mathbf{I} - \mathbf{K}\mathbf{H}^T\right)\mathbf{P}^f, \tag{20.19}$$

then the analysis update for the EnKF is based upon the equation above; however, if we take an ensemble of model states such that the error covariances of the forecasted ensemble mean coincides with the ensemble covariance and then we perform an analysis on each ensemble member, then the error covariance of the analyzed ensemble mean is given by (20.19) [111]. However, the **ensemble covariance** is reduced too much unless the observations are treated as random variables. The reason for this statement is because in the expression for the analysis ensemble covariance there is no analog to the term $\mathbf{K}\mathbb{E}\left[\left(\mathbf{d}-\mathbf{d}^t\right)\left(\mathbf{d}-\mathbf{d}^t\right)^T\right]\mathbf{K}^T = \mathbf{K}\mathbf{O}\mathbf{K}^{\mathbf{T}}$ in (20.19), and as such this leads to spurious correlations because all of the ensemble members are updated with the same observations. This results in the covariance of the analyzed ensemble being

$$\mathbf{P}_e^a = (\mathbf{I} - \mathbf{K}\mathbf{H})\,\mathbf{P}^f\,(\mathbf{I} - \mathbf{K}\mathbf{H})^T, \tag{20.20}$$

where we can see that we have too many $(\mathbf{I} - \mathbf{K}\mathbf{H})^T$ terms. The reason for this is that the original analysis scheme for the EnKF was based upon (20.17a) and (20.17b), while it should have been based upon (20.18a) and (20.18b) [52].

Therefore, the basis of the updated version of the EnKF from [52] is now to treat the observations as random variables by generating an ensemble of observations, where this ensemble is generated from a distribution with a mean that is equal to the first-guess observations and covariance that is equal to $\mathbf{O}$. If we now define the new observations as

$$\mathbf{y}_j = \mathbf{y} + \boldsymbol{\varepsilon}^o, \tag{20.21}$$

where $j = 1, 2, \ldots, n$, then we modify the analysis step of the EnKF as

$$\boldsymbol{\psi}_j^a = \boldsymbol{\psi}_j^f + \mathbf{K}_e\left(\mathbf{y}_j - \mathbf{H}\boldsymbol{\psi}_j^f\right), \tag{20.22}$$

where the ensemble Kalman gain matrix, $\mathbf{K}_e$ is given by

$$\mathbf{K}_e = \mathbf{P}_e^f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_e^f\mathbf{H}^T + \mathbf{O}\right)^{-1}. \tag{20.23}$$

The mean analysis state can be expressed in terms of the mean forecast state, observation, and model representative as

$$\overline{\boldsymbol{\psi}}^a = \overline{\boldsymbol{\psi}}^f + \mathbf{K}_e \left(\overline{\mathbf{y}} - \mathbf{H} \overline{\boldsymbol{\psi}}_j^f \right). \tag{20.24}$$

Given the expressions above, the ensemble analysis error covariance matrix can be shown to be

$$\begin{aligned} \mathbf{P}_e^a &= \mathbb{E}\left[\left(\boldsymbol{\psi}^a - \overline{\boldsymbol{\psi}}^a \right) \left(\boldsymbol{\psi}^a - \overline{\boldsymbol{\psi}}^a \right)^T \right], \\ &= (\mathbf{I} - \mathbf{K}_e \mathbf{H}) \, \mathbf{P}_e^f, \end{aligned} \tag{20.25}$$

where

$$\boldsymbol{\psi}^a - \overline{\boldsymbol{\psi}}^a = (\mathbf{I} - \mathbf{K}_e \mathbf{H}) \left(\boldsymbol{\psi}^f - \overline{\boldsymbol{\psi}}^f \right) + \mathbf{K}_e \left(\mathbf{y} - \overline{y} \right). \tag{20.26}$$

An important feature to note here is that the ensemble of the observations do not affect the update to the ensemble means, since this term is not in (20.24). Another feature to note here is that each ensemble member evolves according to a model such as

$$\boldsymbol{\psi}_j^{k+1} = \boldsymbol{\mathcal{M}} \left(\boldsymbol{\psi}_j^k \right) + d\mathbf{q}_j^k, \tag{20.27}$$

where $d\mathbf{q}$ is a stochastic forcing that represents the model error from a probability distribution with mean zero and covariance $\mathbf{Q}$. The **ensemble covariance** matrix of the error in the model equations is given by

$$\mathbf{Q}_e = \mathbb{E}\left[\left(d\mathbf{q}^k - \overline{d\mathbf{q}}^k \right) \left(d\mathbf{q}^k - \overline{d\mathbf{q}}^k \right)^T \right]. \tag{20.28}$$

The ensemble mean then evolves according to the equation

$$\begin{aligned} \overline{\boldsymbol{\psi}}^{k+1} &= \overline{\boldsymbol{\mathcal{M}} \left(\boldsymbol{\psi}^k \right)}, \\ &= \boldsymbol{\mathcal{M}} \left(\overline{\boldsymbol{\psi}}^k \right) + NLT, \end{aligned} \tag{20.29}$$

where NLT stands for nonlinear term if $\boldsymbol{\mathcal{M}}$ is nonlinear.

However, it was soon shown that perturbing the observations introduces sampling errors [472]. Whitaker and Hamil [472] introduce a new form of the EnKF which would bring about a new set of ensemble-based Kalman filters referred to as the **ensemble square root filters (EnSRFs)**.

20.3 Ensemble Square Root Filters

The motivation for the derivation of the EnSRF, was first to avoid sampling errors caused by perturbing the observations, which was to avoid reduction of the ensemble covariance being too severe, but second to find a formulation of the ensemble Kalman gain matrix such that the analysis error covariance matrix for the ensemble satisfies

$$\mathbf{P}^a = (\mathbf{I} - \mathbf{KH}) \, \mathbf{P}^b \, (\mathbf{I} - \mathbf{KH})^T + \mathbf{KRK}^T = (\mathbf{I} - \mathbf{KH}) \, \mathbf{P}^b. \tag{20.30}$$

Recalling the work of [52]; if all ensemble members are updated with the same observations using the same gain matrix, then the covariance of the analyzed ensemble can be shown to be

$$\mathbf{P}^a = (\mathbf{I} - \mathbf{KH})\,\mathbf{P}^b\,(\mathbf{I} - \mathbf{KH})^T. \tag{20.31}$$

The challenge now is to define an approximation $\widetilde{\mathbf{K}}$ for $\mathbf{K}$ from the ensemble such that when $\widetilde{\mathbf{K}}$ is substituted into (20.31), we obtain a form similar to that on the right-hand side of (20.30).

It is shown in [472] that an expression for $\widetilde{\mathbf{K}}$ that satisfies the required property is given by

$$\widetilde{\mathbf{K}} = \mathbf{P}^b\mathbf{H}^T\left(\left(\sqrt{\mathbf{HP}^b\mathbf{H}^T + \mathbf{R}}\right)^{-1}\right)^T\left(\sqrt{\left(\mathbf{HP}^b\mathbf{H}^T + \mathbf{R}\right)} + \sqrt{\mathbf{R}}\right)^{-1}. \tag{20.32}$$

Note: The square roots of the matrices in (20.32) are not unique, they can be computed in different ways, say with a singular vector decomposition, or through a Cholesky factorization.

If we consider an individual observation, then $\mathbf{HP}^b\mathbf{H}^T$ and $\mathbf{R}$ reduce to scalars and the equation

$$\left(\mathbf{I} - \widetilde{\mathbf{K}}\mathbf{H}\right)\mathbf{P}^b\left(\mathbf{I} - \widetilde{\mathbf{K}}\mathbf{H}\right)\mathbf{P}^b = \left(\mathbf{I} - \widetilde{\mathbf{K}}\mathbf{H}\right)\mathbf{P}^b, \tag{20.33}$$

becomes

$$\frac{\mathbf{HP}^b\mathbf{H}^T}{\mathbf{HP}^b\mathbf{H}^T + \mathbf{R}}\widetilde{\mathbf{K}}\widetilde{\mathbf{K}} - \mathbf{K}\widetilde{\mathbf{K}}^T - \widetilde{\mathbf{K}}\mathbf{K}^T + \mathbf{K}\mathbf{K}^T = \mathbf{0}. \tag{20.34}$$

If we now set $\widetilde{\mathbf{K}} = \alpha\mathbf{K}$, where α is a constant, then it is possible to factorize out the $\mathbf{KK}^T$ term, which results in a quadratic equation for α such that the solution is between 0 and 1 can be shown to be

$$\alpha = \left(1 + \sqrt{\frac{\mathbf{R}}{\mathbf{HP}^b\mathbf{H} + \mathbf{R}}}\right)^{-1}. \tag{20.35}$$

Thus the mean and departure from the mean are updated independently according to

$$\begin{aligned}
\mathbf{K} &= \mathbf{P}^b\mathbf{H}^T\left(\mathbf{HP}^b\mathbf{H}^T + \mathbf{R}\right)^{-1},\\
\overline{\mathbf{x}}^a &= \overline{\mathbf{x}}^b + \mathbf{K}\left(\overline{\mathbf{y}} - \mathbf{H}\overline{\mathbf{x}}^b\right),\\
\mathbf{x}^{a'} &= \mathbf{x}^{b'} - \widetilde{\mathbf{K}}\left(\mathbf{H}\mathbf{x}^{b'}\right).
\end{aligned}$$

We should note that the covariance matrix here is the ensemble-based approximations.

One important feature of the EnSRF is that it processes the observations sequentially, which makes it possible to implement the covariance localization. This improves the analysis while preventing filter divergence in small ensembles. A summary of the EnSRFs up to 2003 can be found in Tippet et al. [439].

In the theory just presented, we saw one of the problems associated with the EnKF, which was rank deficiencies. But there is also another problem associated with the **sampling** of the ensembles. This undersampling can lead to smaller scales not being resolved correctly; we saw this in the VAR subcomponent chapter where we could excite dynamically spurious gravity waves.

While the EnKF does not generate guaranteed balanced initial conditions unless some constraint is applied, there is the problem coming from the undersampling. In the next section we shall introduce a

couple of techniques that have been introduced to compensate for the undersampling because, as with the balance constraints in VAR, it is possible for the filters to *diverge*. These techniques are referred to as **localization and inflation**.

20.3.1 Localization and Inflation

The first appearance of a form of localization is in the paper *Data assimilation using an ensemble Kalman filter technique* by Houtekamer and Mitchell in 1998 [183], where they are running a pair of ensembles of order N so that the covariances calculated for each ensemble is used to assimilate data into the other. As a result of this, the two EnKFs use different ensembles of first-guess fields for the estimation of the forecast errors and the estimation of the analysis error.

The other main difference in the formulation of the EnKF in [183] is that they use a cut-off radius to perform a data selection, which is not usual for the KF, but is used in the SI/OI schemes. Therefore, for each vertical column of analysis points, all the data within a given horizontal distance, r_{max}, are used. This approach is a good method for eliminating observations that are only weakly correlated with the analysis point, which according to [183] would require thousands of ensemble members to accurately resolve.

In Houtekamer and Mitchell's 2001 paper *A sequential ensemble Kalman filter for atmospheric data assimilation* [184], they introduce an ensemble error localization so as to be able to filter the forecast error covariance matrix of the small background-error correlations that come from using too small an ensemble. They achieve this localization by applying a Schur product, which is also referred to as the Hadamard product, which is an element-by-element matrix product, of the covariances of the forecast error that has been calculated from the ensemble and a correlation function that has local support.

Specifically, Houtekamer and Mitchell redefine the ensemble Kalman gain matrix $\mathbf{K}$ as

$$\mathbf{K} = \left(\left(\boldsymbol{\rho} \circ \mathbf{P}^f\right)\mathbf{H}^T\right)\left(\mathbf{H}\left(\boldsymbol{\rho} \circ \mathbf{P}^f\right)\mathbf{H}^T + \mathbf{R}\right)^{-1}, \tag{20.36}$$

where it is possible to interchange the order of the forward interpolation and the Schur product such that

$$\mathbf{k} = \left(\boldsymbol{\rho} \circ \left(\mathbf{P}^f\mathbf{H}^T\right)\right)\left(\boldsymbol{\rho} \circ \left(\mathbf{H}\mathbf{P}^f\mathbf{H}^T\right) + \mathbf{R}\right)^{-1}, \tag{20.37}$$

where

$$\mathbf{P}^f\mathbf{H}^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}\left(\boldsymbol{\psi}_i^f - \overline{\boldsymbol{\psi}}^f\right)\left(\mathbf{H}\boldsymbol{\psi}_i^f - \mathbf{H}\overline{\boldsymbol{\psi}}^f\right),$$

$$\mathbf{H}\mathbf{P}^f\mathbf{H}^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}\left(\mathbf{H}\boldsymbol{\psi}_i^f - \mathbf{H}\overline{\boldsymbol{\psi}}^f\right)\left(\mathbf{H}\boldsymbol{\psi}_i^f - \mathbf{H}\overline{\boldsymbol{\psi}}^f\right).$$

The matrix, $\boldsymbol{\rho}$, is a relatively broad function. For the study presented in [184] the authors use a fifth-order compactly supported piecewise rational function from [152], and is also used in [165]. There are a large suite of different compactly supported two- and three-dimensional correlation functions presented in [152] and the reader is referred to that paper for more details about the different piecewise functions.

Another reason for the localization in either observation space and/or ensemble covariances is to prevent what is referred to as **filter divergence**. The first instance of filter divergence, according to

[165], occurred in [183]. Filter divergence is the process that the ensemble progressively ignores the observations more and more in successive cycles, that leads to a useless ensemble. In [165] the authors state that the cause of the filter divergences is due to using the ensemble to produce reduced rank representation of background error statistics, or, as we have been calling them, forecast error statistics.

In [165] the authors presented two possible sources of the filter divergence that are functions of the background, such that the background at the observation location is adjusted toward the observation but only to an extent consistent with the ratio of the background and observational covariances. The two problems that lead to filter divergence are possibly associated with either the background errors underestimated, so that the observation is comparatively ignored, while if we have that the magnitude of the background error covariances between an observation location and a far removed data point are overestimated due to sampling error, then the posterior probability distribution at the far-removed grid point will be adjusted too much. This implies that the posterior distribution has insufficient probability in the region in phase space near to the true state. For an illustration of these conditions we have a copy of figure 3 from [165] in Fig. 20.3.

While we have localization to remove the undersampled smaller scales from the reduced rank approximation to the error covariance matrices, we have not addressed the fact that the initial conditions could excite spurious gravity modes that could also cause the ensemble members and/or ensemble mean to drift away from the true state. This problem was addressed in a paper titled *Covariance localization and balance in an ensemble Kalman filter* by Dr. Jeffrey Kepert [210]. In [210] Kepert states that the use of covariance localization may introduce imbalances into the system. To address the imbalances, he develops a covariance model from the streamfunction-velocity potential rather than the wind components.

Before the work on the localization of the covariances to ensure balance in the analysis in [210], Mitchell et al. in [303] describe techniques to ensure that the initial perturbations for the ensemble members are in balance.

We move on to consider how to avoid filter divergence through a technique referred to as **inflation**. The notion of inflation appeared in Anderson and Anderson's 1999 paper *A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts* [9]. The inflation technique is to multiply the ensemble approximation to the forecast error covariance matrix by a constant factor γ. The effect of the constant is that it broadens the prior distribution artificially but it prevents the distribution from shrinking. However, we have to be cautious about the size of γ and it is not always possible to know in advance how much of an inflation we require.

In [8] Anderson introduces **adaptive covariance inflation**, which is based upon a hierarchical Bayesian approach, where this algorithm uses the same observations that are used to adjust the ensemble filter estimates of the state, to estimate appropriate values of the covariance inflation. There is a full mathematical derivation of this approach in [8], along with a description of the algorithm to implement it.

In [254] the authors describe a method to use maximum likelihood estimation of inflation factors for the EnKFs. They make reference in [254] and present a method from Wang and Bishop that uses a time-dependent inflation algorithm for the inflation factors [463].

Given these different characteristics of the EnKF we move on to consider three different forms of ensemble Kalman filtering over the next two sections: the **ensemble transform Kalman filter (ETKF)**, the **local ensemble transform Kalman filter (LETKF)**, and the **maximum likelihood ensemble filter (MLEF)**.

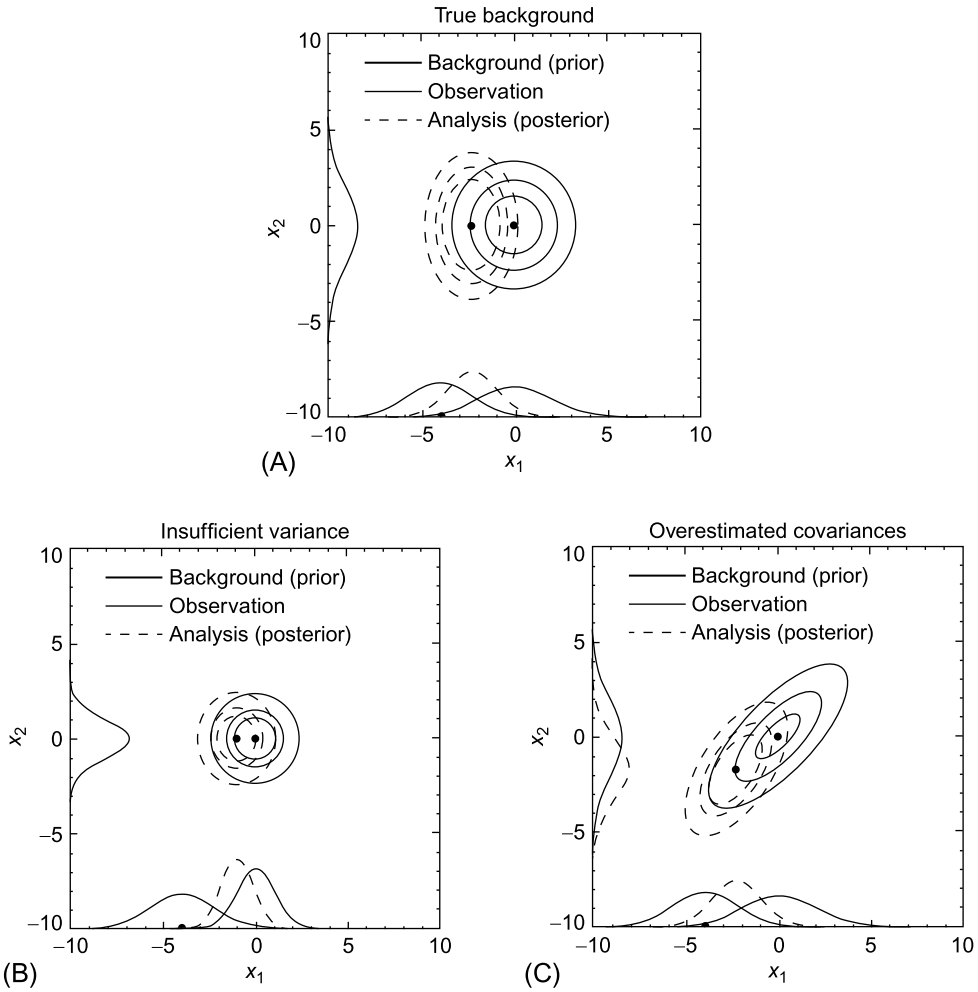


FIGURE 20.3

Copy of the figure showing the effects of incorrect sampling with ensembles from Hamill, T. M., Whitaker, J. S., and Snyder, C. (2001). Distance-Dependent Filtering of Background Error Covariance Estimates in an Ensemble Kalman Filter, Monthly Weather Review, 129(11), 2776-2790. **© American Meteorological Society. Used with permission.**

20.4 Ensemble and Local Ensemble Transform Kalman Filter

In this section we shall introduce two versions of the EnKF that are used quite extensively. The first filter is the ETKF. The ETKF first appeared in [40]; however, here we summarize the derivation from Bishop et al. [38], where it is a bit easier to follow.

20.4.1 ETKF

We start with the analysis error covariance matrix from the EnKF:

$$\mathbf{P}_t^a = \frac{1}{K-1}\sum_{k=1}^{K}\left(\mathbf{x}_k\left(t_i\right)-\overline{\mathbf{x}}\left(t_i\right)\right)\left(\mathbf{x}_k\left(t_i\right)-\overline{\mathbf{x}}\left(t_i\right)\right)^T = \mathbf{X}_t\mathbf{X}_t^T, \tag{20.38}$$

where the columns of $\mathbf{X}$ are given by $\frac{\mathbf{x}_k(t_i)-\overline{\mathbf{x}}(t_i)}{\sqrt{K-1}}$. Therefore, according to (20.38), the forecast error matrix at t_{i+1} is given by

$$\mathbf{P}_{t+1}^f = \mathbf{Z}_{t+1}\mathbf{Z}_{t+1}^T = \mathbf{X}_{t+1}\mathbf{T}_0\mathbf{T}_0^T\mathbf{X}_{t+1}^T, \tag{20.39}$$

where at the initial time $\mathbf{T}_0$ is equal to the identity $\mathbf{I}$. Thus at any later data assimilation time t_{i+m}, we have

$$\mathbf{P}_{t+m}^f = \mathbf{Z}_{i+m}\mathbf{Z}_{i+m}^T = \mathbf{X}_{i+m}\mathbf{T}_{t+m-1}\mathbf{T}_{t+m-1}^T\mathbf{X}_{i+m}, \tag{20.40}$$

where $\mathbf{T}_{t+m-1}$ is a $K \times K$ transformation matrix that is generally not equal to the identity matrix.

The first goal of [38] is to be able to write the ensemble-based analysis error covariance matrix in the form

$$\mathbf{P}^a = \mathbf{P}^f - \mathbf{P}^f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}^f\mathbf{H}^T+\mathbf{R}\right)^{-1} = \mathbf{Z}^f\mathbf{T}\mathbf{T}^T\mathbf{Z}^{f^T}, \tag{20.41}$$

for the transformation matrix, $\mathbf{T}$, given that the forecast error covariance matrix $\mathbf{P}^f = \mathbf{Z}^f\mathbf{Z}^{f^T}$.

The next goal of the EnKFs is to avoid the size and ill-conditioning problems associated with the reduce rank approximations. To achieve this the ETKF introduces a normalized observation operator, $\widetilde{\mathbf{H}} = \mathbf{R}^{-\frac{1}{2}}\mathbf{H}$, so that

$$\begin{aligned}\mathbf{P}^f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}^f\mathbf{H}^T+\mathbf{R}\right)^{-1}\mathbf{H}\mathbf{P}^f &= \mathbf{P}^f\mathbf{H}^T\left(\mathbf{R}^{-\frac{1}{2}}\left(\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}^f\mathbf{H}^T\mathbf{R}^{-\frac{1}{2}}+\mathbf{I}\right)\mathbf{R}^{-\frac{1}{2}}\right)^{-1}\mathbf{H}\mathbf{P}^f,\\ &= \mathbf{P}^f\widetilde{\mathbf{H}}^T\left(\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T+\mathbf{I}\right)^{-1}\widetilde{\mathbf{H}}\mathbf{P}^f,\end{aligned} \tag{20.42}$$

where $\mathbf{I}$ is of the dimensions of the number of observations. We now apply a eigenvector decomposition to (20.42) using the property that the eigenvectors of $\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T$ are equivalent to those of $\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T+\mathbf{I}$, which implies that

$$\left(\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T+\mathbf{I}\right)^{-1} = \mathbf{E}^c\left(\mathbf{\Gamma}^c+\mathbf{I}\right)^{-1}\mathbf{E}^c, \tag{20.43}$$

where the p columns of $\mathbf{E}^c$ contain the complete set of orthonormal eigenvectors of $\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T$ and the diagonal matrix $\mathbf{\Gamma}$ contains the corresponding eigenvalues. We only require the eigenvectors not in the right null space of $\mathbf{P}^f\mathbf{H}^T$. Since

$$\mathbf{P}^f\widetilde{\mathbf{H}}^T = \mathbf{Z}^f\mathbf{Z}^{f^T}\widetilde{\mathbf{H}}^T, \tag{20.44}$$

then only the eigenvectors of $\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T$ that contribute to the analysis error covariance matrix defined by (20.41) are those that can be written as a linear combination of the column vectors of $\widetilde{\mathbf{H}}\mathbf{Z}^f$. However,

we should note that not all of the eigenvectors are linearly independent because the sum of the K ensemble perturbations from which they are derived is equal to zero. Therefore, we seek the set of eigenvectors that are linearly independent to form the vector space.

To obtain the subset of eigenvectors $\mathbf{E}$ of the complete set $\mathbf{E}^c$ that can be written as linear combinations of the columns of $\widetilde{\mathbf{H}}\mathbf{Z}^f$, let the matrix $\mathbf{C}$ contain the orthonormal eigenvectors of $\mathbf{Z}^{f^T}\widetilde{\mathbf{H}}^T\widetilde{\mathbf{H}}\mathbf{Z}^f$, such that

$$\left(\mathbf{Z}^{f^T}\widetilde{\mathbf{H}}^T\widetilde{\mathbf{H}}\mathbf{Z}^f\right)\mathbf{C} = \mathbf{C}\boldsymbol{\Gamma}, \tag{20.45}$$

where $\boldsymbol{\Gamma}$ is a diagonal matrix that contains the eigenvalues γ_{kk} of $\mathbf{Z}^{f^T}\widetilde{\mathbf{H}}^T\widetilde{\mathbf{H}}\mathbf{Z}^f$.

We know that there are going to be eigenvalues that are equal to zero due to the fact that not all of the eigenvectors are linearly independent. This implies that for $k > k_c$, we have zero eigenvalues and so $\mathbf{Z}^f\mathbf{c}_k = \mathbf{0}$. If we now let $\widehat{\boldsymbol{\Gamma}}$ be from $\boldsymbol{\Gamma}$ by setting all of the zero eigenvalues in $\boldsymbol{\Gamma} = 1$, then this results in

$$\mathbf{E} = \widetilde{\mathbf{H}}\mathbf{Z}^f\mathbf{C}\widehat{\boldsymbol{\Gamma}}^{-\frac{1}{2}}. \tag{20.46}$$

Through the arguments presented in [38] we see that the first k_c columns of $\mathbf{E}$ are orthonormal and the remaining columns are zero vectors, and that the matrix $\mathbf{E}$ contains all of the singular vectors in the vector subspace spanned by the vectors corresponding to the columns of $\widetilde{\mathbf{H}}\mathbf{Z}^f$.

After some manipulations, see [38] for details, we arrive at the three main ETKF equations:

$$\mathbf{P}^f\widetilde{\mathbf{H}}^T\left(\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T + \mathbf{I}\right)^{-1}\widetilde{\mathbf{H}}\mathbf{P}^f = \mathbf{Z}\left(t^a\right)\mathbf{C}\boldsymbol{\Gamma}\left(\boldsymbol{\Gamma} + \mathbf{I}\right)^{-1}\mathbf{C}^T\mathbf{Z}^{f^T}\left(t^a\right), \tag{20.47}$$

$$\mathbf{x}^a(t) - \mathbf{x}^f(t) = \mathbf{M}\left(t, t^a\right)\mathbf{P}^f\widetilde{\mathbf{H}}^T\left(\widetilde{\mathbf{H}}\mathbf{P}^f\widetilde{\mathbf{H}}^T + \mathbf{I}\right)^{-1}\left(\mathbf{R}^{-\frac{1}{2}}\mathbf{y} - \widetilde{\mathbf{H}}\mathbf{x}^f\left(t^a\right)\right), \tag{20.48}$$

$$\mathbf{x}^a(t) - \overline{\mathbf{x}}^f(t) = \mathbf{Z}^f(t)\mathbf{C}\boldsymbol{\Gamma}^{\frac{1}{2}}\left(\boldsymbol{\Gamma} + \mathbf{I}\right)^{-1}\mathbf{E}^T\left(\mathbf{R}^{-\frac{1}{2}}\mathbf{y} - \widetilde{\mathbf{H}}\mathbf{x}^f\left(t^a\right)\right), \tag{20.49}$$

where $\mathbf{M}$ is the numerical model and $\mathbf{y}$ is the vector of observations.

Therefore, we can see that the ETKF is a singular vector decomposition-based approach to avoid the undersampling affecting the ensemble spread. For more details on how (20.47)–(20.49) are obtained, see [38].

20.4.2 LETKF

We now more on to consider variation of the ETKF, referred to as the **LETKF** or **Local Ensemble Transform Kalman Filter**.

The LETKF was developed, and is still maintained, at the University of Maryland, and is used in many different geophysical applications. We shall now summarize the key components of the LETKF from Ott et al. [324].

Ott et al. (2004) first present an outline of the algorithm with seven steps (we have adapted the description in Algorithm 20.1 of the steps to be applicable to all geophysical fields).

We now summarize what is meant by the local domain: the LETKF performs the analysis step of the ETKF in a local region, hence the L part, but in [324] the authors state that a model state of a geophysical process is given a vector field $\mathbf{x}(\mathbf{r}, t)$ where $\mathbf{r}$ is a two-dimensional grid point coordinate. An important comment that they make about $\mathbf{x}$ in [324] is that in data assimilation this field is treated as

Algorithm 20.1 LETKF Algorithm

1. Advance the analysis ensemble of global geophysical states to the next analysis time. This is a new background ensemble of global geophysical states.
2. Associate a local region with each grid point and, for each local region and each member of the background ensemble, form vectors of geophysical state information in that local region.
3. For each local ensemble member vector obtained from the previous step, calculate its perturbation from the background ensemble mean, and project these perturbations on to a lower dimensional subspace that best represents the ensemble in that region.
4. Perform the data assimilation in each of the local low dimensional subspaces, obtaining analysis mean and covariance in each local region.
5. From the local analysis mean and covariance, obtain a suitable local analysis ensemble of local geophysical states.
6. Use the local analyses obtained in step 5 to form a new global analysis ensemble.
7. return to 1.

a random variable that is characterized by a probability distribution. The characteristics of $\mathbf{x}$ is updated in two ways: (1) it is evolved according to the model dynamics; and (2) it is modified periodically to take into account recent observations of the geophysical state.

Now at each point the LETKF introduces *local vectors* $\mathbf{x}_{mn}$ of the information $\mathbf{x}\left(\mathbf{r}_{m+m',n+n'}\right)$ for $-l \leq m'$ and $n' \leq l$. That is to say that $\mathbf{x}_{mn}$ specifies the geophysical state within a $(2l+1)$ by $(2l+1)$ path of grid points centered at $\mathbf{r}_{mn}$. If we let u represent the dimensionality of $\mathbf{x}$ at $\mathbf{r}_{mn}$, then the dimensionality of the local patch is $(2l+1)^2 u$.

Next we consider local vectors obtained from the model as forecasts and denote these by $\mathbf{x}^b_{mn}$. Now we let $F_{mn}\left(\mathbf{x}^b_{mn}\right)$ be an approximation to a Gaussian probability density function as

$$F_{mn}\left(\mathbf{x}^b_{mn}\right) \sim \exp\left\{-\frac{1}{2}\left(\mathbf{x}^b_{mn}-\overline{\mathbf{x}}^b_{mn}\right)^T\left(\mathbf{P}^b_{mn}\right)^{-1}\left(\mathbf{x}^b_{mn}-\overline{\mathbf{x}}^b_{mn}\right)\right\}, \tag{20.50}$$

where $\mathbf{P}^b_{mn}$ and $\overline{\mathbf{x}}^b_{mn}$ are the local background error covariance matrix and *most probable state* associated with $F_{mn}\left(\mathbf{x}^b_{mn}\right)$.

We add here a word of caution over terminology being used in Ott et al. to describe $\overline{\mathbf{x}}_{mn}$; as we mentioned in the summary of [111] this state is only the most probable if we are optimizing a Gaussian distribution. Whereas for a lognormal distribution, the approximation used for the most probable state would have to be applied to the logarithm of the field, but that would be the median in lognormal space, which is the unbiased state, but not the most probable.

Returning to [324], we have the same situation as with the ETKF of the background error covariance matrix $\mathbf{P}^b_{mn}$ that has a $(2l+1)^2 u - k$ null space, where $k = \text{rank}\left(\mathbf{P}^b_{mn}\right)$.

The local ensemble mean state is not always the most probable, especially when it is calculated as

$$\overline{\mathbf{x}}^b_{mn} = \frac{1}{k'+1}\sum_{i=1}^{k'+1}\mathbf{x}^{b(i)}_{mn}, \tag{20.51}$$

where $k'+1$ is the number of ensemble members, where $k'+1 \geq k \geq 1$ and $\mathbf{x}_{mn}^{b(i)}$ is the local vector for the patch centered at $\mathbf{r}_{mn}$ and the ensemble member $\mathbf{x}^{b(i)}$.

To obtain an expression for the local background error covariance matrix $\mathbf{P}_{mn}^{b}$, that is used in the analysis, we first consider a matrix $\mathbf{P}_{mn}^{b'}$ defined as

$$\mathbf{P}_{mn}^{b'} = \mathbf{X}_{mn}^{b}\mathbf{X}_{mn}^{b^T}, \tag{20.52}$$

given that

$$\mathbf{X}_{mn}^{b} \equiv \frac{1}{\sqrt{k'}}\left[\delta\mathbf{x}_{mn}^{b(1)}, \delta\mathbf{x}_{mn}^{b(2)}, \ldots, \delta\mathbf{x}_{mn}^{b(k'+1)}\right], \tag{20.53}$$

$$\delta\mathbf{x}_{mn}^{b(i)} = \mathbf{x}_{mn}^{b(i)} - \overline{\mathbf{x}}_{nm}^{b}. \tag{20.54}$$

Now let the eigenvalues of the matrix $\mathbf{P}_{mn}^{b'}$ be denoted by $\lambda_{nm}^{(j)}$. Because $\mathbf{P}_{mn}^{b}$ is a symmetric matrix, it has k' orthonormal eigenvectors $\left\{\mathbf{u}_{mn}^{(j)}\right\}$ that correspond to the k' eigenvalues, thus

$$\mathbf{P}_{mn}^{b'} = \sum_{j=1}^{k'} \lambda_{mn}^{(j)}\mathbf{u}_{mn}^{(j)}\left(\mathbf{u}_{mn}^{(j)}\right)^T. \tag{20.55}$$

Because the number of ensemble members is going to be much smaller than the dimensions of the problem in the local region, the summation in (20.55) is truncated at k. For the purpose of subsequent computation, we consider the coordinate system for the k-dimensional space $\mathbb{S}_{mn}$ determined by the basis vectors $\mathbf{u}_{\mathbf{mn}}^{(\mathbf{j})}$. These coordinates are referred to as the internal coordinate system for $\mathbb{S}_{mn}$. To change between the internal coordinates and those of the local space, Ott et al. introduce a $(2l+1)^2\,u \times k$ matrix given by

$$\mathbf{Q}_{mn} \equiv \left(\mathbf{u}_{mn}^{(1)}, \mathbf{u}_{mn}^{(2)}, \ldots, \mathbf{u}_{mn}^{(k)}\right). \tag{20.56}$$

The matrix $\mathbf{Q}_{mn}$ is a projection on to the subspace $\mathbb{S}_{mn}$ and as such for a given vector $\mathbf{z}$, which is a $(2l+1)^2\,u$ dimensional column vector, $\widehat{\mathbf{z}}$ is a k dimensional column vector where $\widehat{\mathbf{z}} = \mathbf{Q}_{nm}\mathbf{z}$. As a result of this projection, we should note that the new vector is in terms of the internal coordinate system. We can also project the linearized, or linear, numerical model matrix $\mathbf{M}$ into this local coordinate system by

$$\widehat{\mathbf{M}} = \mathbf{Q}_{mn}^{T}\mathbf{M}\mathbf{Q}_{mn}. \tag{20.57}$$

As the columns of the matrix $\mathbf{Q}_{mn}$ are the orthonormal eigenvector, we have that $\mathbf{Q}_{mn}^{T}\mathbf{Q}_{mn} = \mathbf{I}$, whereas $\mathbf{Q}_{nm}\mathbf{Q}_{mn}^{T}$ represents projections on $\mathbb{S}_{mn}$, and has a null space $\overline{\mathbb{S}}_{mn}$.

We can now write the vector $\mathbf{z}$ in terms of components that are in $\mathbb{S}_{mn}$ and $\overline{\mathbb{S}}_{mn}$ as

$$\mathbf{z} = \mathbf{z}^{\parallel} + \mathbf{z}^{\perp}, \tag{20.58}$$

$$\mathbf{z} = \mathbf{\Lambda}_{nm}^{\parallel}\mathbf{z} = \mathbf{Q}_{mn}\widehat{\mathbf{z}}, \quad \mathbf{z}^{\perp} = \mathbf{\Lambda}^{\perp}\mathbf{z}, \tag{20.59}$$

where $\mathbf{z}^{\parallel} \in \mathbb{S}_{mn}$, while $\mathbf{z}^{\perp} \in \overline{\mathbb{S}}_{mn}$ and

$$\mathbf{\Lambda}_{mn}^{\parallel} \equiv \mathbf{Q}_{mn}\mathbf{Q}_{mn}^{T}, \quad \mathbf{\Lambda}_{mn}^{\perp} \equiv \mathbf{I} - \mathbf{Q}_{mn}\mathbf{Q}_{mn}^{T}. \tag{20.60}$$

As a result of these projections, we have that

$$\mathbf{M} = \mathbf{Q}_{mn}\widehat{\mathbf{M}}\mathbf{Q}_{mn}^{T}, \tag{20.61a}$$

$$\widehat{\mathbf{P}}_{mn}^{b} = \text{diag}\left[\lambda_{mn}^{1}, \lambda_{mn}^{2}, \ldots, \lambda_{mn}^{k}\right], \tag{20.61b}$$

and as such the projected background error covariance matrix is easy to invert.

The next step that Ott et al. [324] move on to is the assimilation step, which is to minimize the incremental 3D VAR cost function, but projected into ensemble space. They linearize the nonlinear observation operator about the mean ensemble background state $\overline{\mathbf{x}}_{mn}^{b}$, where this state is assumed to be quite close to the true state, that they denote as $\mathbf{x}_{mn}^{a}$, which is the **local analysis**. We shall not go into all of the details about the minimization but will present the important equations:

$$\mathbf{h}\left(\mathbf{x}_{mn}^{a}\right) \approx \mathbf{h}\left(\overline{\mathbf{x}}_{mn}^{b}\right) + \mathbf{H}_{mn}\Delta\mathbf{x}_{mn}^{a}, \tag{20.62}$$

$$\Delta\mathbf{x}_{mn}^{a} = \mathbf{x}_{mn}^{a} - \overline{\mathbf{x}}_{mn}^{b}, \tag{20.63}$$

$$\begin{aligned} J\left(\Delta\widehat{\mathbf{x}}_{mn}^{a}\right) = &\frac{1}{2}\left(\Delta\widehat{\mathbf{x}}_{mn}^{a}\right)^{2}\left(\widehat{\mathbf{P}}_{mn}^{b}\right)^{-1}\Delta\widehat{\mathbf{x}}_{mn}^{a} \\ &+ \frac{1}{2}\left(\widehat{\mathbf{H}}_{mn}\Delta\widehat{\mathbf{x}}_{mn}^{a} + \mathbf{h}_{mn}\left(\overline{\mathbf{x}}_{mn}^{b}\right) - \mathbf{y}_{mn}\right)^{T}\mathbf{R}_{mn}^{-1}\left(\widehat{\mathbf{H}}_{mn}\Delta\widehat{\mathbf{x}}_{mn}^{a} + \mathbf{h}_{mn}\left(\overline{\mathbf{x}}_{mn}^{b}\right) - \mathbf{y}_{mn}\right), \end{aligned} \tag{20.64}$$

where the superscript $\hat{\cdot}$ represents that matrix has been projected by $\mathbf{Q}_{mn}$ into the $\mathbb{S}_{mn}$ subspace.

Now if minimizing the cost function in (20.64), then the state $\widehat{\mathbf{x}}_{mn}^{a}$ is the most probable state, which is equal to

$$\widehat{\mathbf{x}}_{mn}^{a} = \widehat{\mathbf{P}}_{mn}^{b}\widehat{\mathbf{H}}_{mn}^{T}\mathbf{R}_{mn}^{-1}\left(\mathbf{y}_{mn} - \mathbf{h}_{mn}\left(\overline{\mathbf{x}}_{mn}^{b}\right)\right), \tag{20.65}$$

where the analysis covariance matrix, which is the inverse of the Hessian of (20.64), is given by

$$\widehat{\mathbf{P}}_{mn}^{a} = \left(\left[\widehat{\mathbf{P}}_{mn}^{b}\right]^{-1} + \widehat{\mathbf{H}}_{mn}^{T}\mathbf{R}_{mn}^{-1}\widehat{\mathbf{H}}_{mn}^{T}\right)^{-1} = \widehat{\mathbf{P}}_{mn}^{b}\left(\mathbf{I} + \widehat{\mathbf{H}}_{mn}^{T}\mathbf{R}_{mn}^{-1}\widehat{\mathbf{H}}_{mn}^{T}\widehat{\mathbf{P}}_{mn}^{b}\right)^{-1}. \tag{20.66}$$

To return to the local state space representation we have that

$$\widehat{\mathbf{x}}_{mn}^{a} = \mathbf{Q}_{mn}\Delta\widehat{\mathbf{x}}_{mn}^{a} + \overline{\mathbf{x}}_{mn}^{b}. \tag{20.67}$$

The remaining task is to use the local analysis information of the mean state and the covariance matrices to obtain an ensemble of global analysis fields, which will be used as the initial conditions for the next numerical model run, which will then become the background fields for the next assimilation cycle.

Therefore, let the $\left(k'+1\right)$ local analyses vectors be defined as

$$\mathbf{x}_{mn}^{a(i)} = \overline{x}_{mn}^{a} + \delta\mathbf{x}_{mn}^{a(i)}. \tag{20.68}$$

By using equations (20.58) and (20.59), then it is possible to write

$$\delta\mathbf{x}_{mn}^{a(i)} = \delta\mathbf{x}_{mn}^{a(i)\parallel} + \delta\mathbf{x}_{mn}^{a(i)\perp} = \mathbf{Q}_{mn}\delta\mathbf{x}_{mn}^{a(i)} + \delta\mathbf{x}_{mn}^{a(i)\perp}. \tag{20.69}$$

Now let

$$\delta\mathbf{x}_{mn}^{a(i)\perp} = \delta\mathbf{x}_{mn}^{b(i)\perp} = \mathbf{\Lambda}_{mn}^{\perp}\delta\mathbf{x}_{mn}^{b(i)}, \tag{20.70}$$

because the analysis uses the observations only to reduce the variance in the space $\mathbb{S}_{mn}$, leaving the variance in $\overline{\mathbb{S}}_{mn}$ unchanged.

If we now combine (20.68)–(20.70), then we obtain

$$\mathbf{x}_{mn}^{a(i)} = \overline{x}_{mn}^{a} + \mathbf{Q}_{mn}\delta\mathbf{x}_{mn}^{a(i)} + \mathbf{\Lambda}_{mn}^{\perp}\delta\mathbf{x}_{mn}^{b(i)}. \tag{20.71}$$

It is stated in [324] that we require

$$\sum_{i=1}^{k'+1} \delta\mathbf{x}_{mn}^{a(i)} = \mathbf{0}, \tag{20.72a}$$

$$\Rightarrow \sum_{i=1}^{k'+1} \delta\mathbf{x}_{mn}^{b(i)} = \mathbf{0}, \tag{20.72b}$$

$$\Rightarrow \sum_{i=1}^{k'+1} \delta\mathbf{x}_{mn}^{a(i)\|} = \mathbf{Q}_{mn} \sum_{i=1}^{k'+1} \delta\widehat{\mathbf{x}}_{mn}^{a(i)} = \mathbf{0}, \tag{20.72c}$$

$$\Rightarrow \sum_{i=1}^{k'+1} \delta\widehat{\mathbf{x}}_{mn}^{a(i)} = \mathbf{0}. \tag{20.72d}$$

We also have that

$$\widehat{\mathbf{P}}_{mn}^{a} = \frac{1}{k'} \sum_{i=1}^{k'+1} \delta\widehat{\mathbf{x}}_{mn}^{a(i)} \left(\delta\widehat{\mathbf{x}}_{mn}^{a(i)}\right)^{T}. \tag{20.73}$$

The next part of [324] is concerned with determining the ensemble of local analysis perturbations. The motivation for this component of the LETKF is associated with balance in the initial conditions for the atmosphere. The authors wish for the analysis ensemble vectors $\delta\widehat{\mathbf{x}}_{mn}^{a(i)}$ to be linearly related to the background ensemble $\delta\widehat{\mathbf{x}}_{mn}^{b(i)}$ for $i = 1, 2, \ldots, k'+1$ as

$$\widehat{\mathbf{X}}_{mn}^{a} = \widehat{\mathbf{X}}_{mn}^{b}\mathbf{Y}_{mn}, \tag{20.74}$$

where $\mathbf{Y}_{mn}$ is a $\left(k'+1\right) \times \left(k'+1\right)$ array and that

$$\widehat{\mathbf{X}}_{mn}^{a,b} = \frac{1}{k'}\left[\delta\widehat{\mathbf{x}}_{mn}^{a,b(1)}, \delta\widehat{\mathbf{x}}_{mn}^{a,b(2)}, \ldots, \delta\widehat{\mathbf{x}}_{mn}^{a,b(k'+1)}\right]. \tag{20.75}$$

After referring to the work in [38], along with the work on square root filters from [439], the authors decide to apply a square root approximation to the ensemble space analysis background and analysis error covariance matrices. Thus Ott et al. show that the optimal choice for $\mathbf{Y}_{nm}$ is

$$\mathbf{Y}_{mn} = \left(\mathbf{I} + \mathbf{X}_{mn}^{b^T}\left(\widehat{\mathbf{P}}_{mn}^{b}\right)^{-1}\left(\widehat{\mathbf{P}}_{mn}^{a} - \widehat{\mathbf{P}}_{mn}^{b}\right)\left[\widehat{bfP}_{mn}^{b}\right]^{1}\mathbf{X}_{mn}^{b}\right)^{\frac{1}{2}}. \tag{20.76}$$

As we have seen from the derivation above, the idea of reducing the dimension of the space within which you wish to perform the analysis in is a useful tool. With the LETKF they used local areas to perform the minimization of a maximum likelihood Gaussian-based cost function, but then update the

moments of the analysis error covariance matrix through the Kalman filter equations. We now consider an alternative approach to this, where the starting point is to take a global cost function and project it into ensemble space to minimize. This technique is referred to as the **Maximum Likelihood Ensemble Filter** or **MLEF**.

20.5 Maximum Likelihood Ensemble Filter

A problem associated with the ensemble-based methods presented so far is the question of whether or not the optimal number of ensemble members required to obtain a desired order of accuracy in our solutions is dependent on the **sampling** structure. Sampling here is meant in the statistical sense where a set of data are taken and used to approximate the moments of the distribution of the random variable. Most of the ensemble filters so far have employed an approximation to the true mean; the LETKF being the exception, through an additive average over all the states in the different ensemble members. From this a sample approximation is made to the second-order moment, variance/covariance.

As we mentioned at the end of the last section, the MLEF is a global version of certain parts of the LETKF, in that it minimizes a cost function for the mode, hence the maximum likelihood in the name, but in ensemble space similar to the LETKF, where the MLEF tries to keep as much of the nonlinearity in the numerical model and observation operators as possible. The MLEF is seen as part of the square-root filters family and has some features in common with the ETKF; a detailed explanation about the difference between the MLEF and the ETKF can be found in [507].

We now present a summary of the steps involved in the MLEF from Fletcher and Zupanski [138]. The first appearance of the MLEF is in Zupanski [507], which is before the introduction of the initiation step, note this is not an initialization technique that we present first; the initial perturbations this approach generate (see below for the method) are more structured and less correlated, which caused a problem in the initial formulation. See Zupanski et al. [509] for more details.

In an ensemble filter, it is desirable to have ensemble members that do not excite spurious gravity modes in the model, or to have perturbations that are unphysical. Both approaches could cause filter divergence.

A more dynamically based approach to generate the seeds for the ensemble members is described in [509]. This technique, is based on using the Kardar-Parisi-Zhang (KPZ) equation:

$$\frac{\partial \phi}{\partial t} = \frac{\partial^2 \phi}{\partial x^2} + \left(\frac{\partial \phi}{\partial x} \right)^2 + \zeta \left(x, t \right), \tag{20.77}$$

where ϕ is the perturbation and ζ is a random forcing.

The reason for choosing this equation is that it is related to the dynamic localization of the Lyapunov vectors. These can be viewed as the exponentials of the roughened interface [337]. The KPZ equation generates spatially sparse, uncorrelated, random perturbations. These perturbations are then smoothed through a space-limited compactly supported function [152]. The correlation length should be realistic to the spatial scales of the associated dynamics being modeled. The approach above generates spatial perturbations that are smoothed to a predefined correlation length. A full, more detailed study of this approach, with the shallow water equations model initialized with test case 5 from [478], can be found in [509]. We should mention that the work presented in [138] was with the global spherical coordi-

nate shallow water equations model; this initiation process does not yield the Lyapunov vectors of the shallow water equations model.

Once the perturbations are generated and smoothed they are added to the ensemble members at a predetermined time before the start of the first assimilation cycle. The ensemble members are then run to initialization time. At this time the initial analysis square root error covariance matrix is calculated.

20.5.1 Forecast Step

The MLEF comprises of two different stages. The first stage is the forecast step and is concerned with the evolution of the forecast error covariances. The starting point for this step is from the evolution equation of the discrete version of the Kalman filter [198]. Therefore, we have that

$$\mathbf{P}^f(k) = \boldsymbol{\mathcal{M}}_{k-1,k}\mathbf{P}_a(k-1)\boldsymbol{\mathcal{M}}^T_{k-1,k} + \mathbf{Q}(k-1), \tag{20.78}$$

where $\mathbf{P}^f$ is the forecast error covariance matrix, k is the time index, $\boldsymbol{\mathcal{M}}$ is the **nonlinear** model evolution operator, and $\mathbf{Q}$ is the model error matrix which is assumed to be Gaussian distributed. For the purpose of this work this is assumed to be zero.

A factorization of $\mathbf{P}^f$ into a square root form can be defined as

$$\mathbf{P}^f = \boldsymbol{\mathcal{M}}\mathbf{P}_a\boldsymbol{\mathcal{M}}^T = \left(\boldsymbol{\mathcal{M}}\mathbf{P}_a^{\frac{1}{2}}\right)\left(\boldsymbol{\mathcal{M}}\mathbf{P}_a^{\frac{1}{2}}\right)^T = \mathbf{P}_f^{\frac{1}{2}}\mathbf{P}_f^{\frac{T}{2}}. \tag{20.79}$$

The structure of the square-root analysis error covariance matrix, $\mathbf{P}_a^{\frac{1}{2}}$, is

$$\mathbf{P}_a^{\frac{1}{2}} = \begin{pmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \dots & \mathbf{p}_S \end{pmatrix} \quad \text{where } \mathbf{p}_i = \begin{pmatrix} p_{1,i} \\ p_{2,i} \\ \vdots \\ p_{N,i} \end{pmatrix}, \tag{20.80}$$

where N is the number of state variables, and S is the number of ensemble members, assuming $S \ll N$.

Upon expanding (20.80), the square root forecast error covariance matrix, $\mathbf{P}_f^{\frac{1}{2}}$, can be expressed as

$$\begin{aligned} \mathbf{P}_f^{\frac{1}{2}} &= \begin{pmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \dots & \mathbf{b}_S \end{pmatrix}, \\ \mathbf{b}_i &= \boldsymbol{\mathcal{M}}\left(\mathbf{x}_{k-1} + \mathbf{p}_i\right) - \boldsymbol{\mathcal{M}}\left(\mathbf{x}_{k-1}\right) \approx \boldsymbol{\mathcal{M}}\mathbf{p}_i, \end{aligned} \tag{20.81}$$

where $\mathbf{x}_{k-1}$ is the analysis state from the previous assimilation cycle, which is found from the posterior analysis PDF [259]. Therefore the MLEF evolves the square root analysis error covariance matrix through the ensemble members.

20.5.2 Analysis Step

The second step in the MLEF is the analysis step which involves solving a nonlinear cost function, similar to that of [259], based upon a Gaussian assumption for the background and observational errors.

The associated cost function is defined in terms of $\mathbf{P}_f$, although this matrix is never calculated or stored in the process of the filter. This results in

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x}-\mathbf{x}_b)^T \mathbf{P}_f^{-1}(\mathbf{x}-\mathbf{x}_b) + \frac{1}{2}(\mathbf{y}-\mathbf{h}(\mathbf{x}))^T \mathbf{R}^{-1}(\mathbf{y}-\mathbf{h}(\mathbf{x})), \tag{20.82}$$

where $\mathbf{y}$ is the vector of observations, $\mathbf{h}$ is the nonlinear observation operator, $\mathbf{R}$ is the observational covariance matrix, and $\mathbf{x}_b$ is a background state, such that $\mathbf{x}_b = \mathcal{M}(\mathbf{x}_{k-1})$.

To find the minimum of (20.82), we introduce a change of variable through a Hessian preconditioner, defined by

$$\mathbf{x}-\mathbf{x}_b = \mathbf{P}_f^{\frac{1}{2}}(\mathbf{I}+\mathbf{C})^{-\frac{T}{2}}\boldsymbol{\xi}, \tag{20.83}$$

where $\boldsymbol{\xi}$ is our vector of control variables, defined in ensemble subspace, and $\mathbf{C}$ is the Hessian matrix of (20.82), which is

$$\mathbf{C} = \mathbf{P}_f^{\frac{T}{2}}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}_f^{\frac{1}{2}} = \left(\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}_f^{\frac{1}{2}}\right)^T\left(\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}_f^{\frac{1}{2}}\right), \tag{20.84}$$

where $\mathbf{H}$ is the Jacobian matrix of $\mathbf{h}$ evaluated at $\mathbf{x}_b$.

It may be the case that the observation operator is nonlinear, difficult to differentiate analytically, or even discontinuous. To overcome this problem we use information from $\mathbf{P}^f$ to approximate the square root of $\mathbf{C}$, componentwise, as

$$\mathbf{z}_i = \left(\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}_f^{\frac{1}{2}}\right)_i = \mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{b}_i \approx \mathbf{R}^{-\frac{1}{2}}(\mathbf{h}(\mathbf{x}+\mathbf{b}_i)-\mathbf{h}(\mathbf{x})). \tag{20.85}$$

It is shown in [510] that the MLEF is a non-differentiable minimization algorithm due to these types of approximations.

A new matrix $\mathbf{Z}$ is now defined such that

$$\mathbf{Z} = \begin{pmatrix} \mathbf{z}_1 & \mathbf{z}_2 & \cdots & \mathbf{z}_S \end{pmatrix}. \tag{20.86}$$

The definition above allows $\mathbf{C}$ to be written as $\mathbf{C} = \mathbf{Z}^T\mathbf{Z}$, i.e.,

$$\mathbf{C} = \begin{pmatrix} \mathbf{z}_1^T\mathbf{z}_1 & \mathbf{z}_1^T\mathbf{z}_2 & \cdots & \mathbf{z}_1^T\mathbf{z}_S \\ \mathbf{z}_2^T\mathbf{z}_1 & \mathbf{z}_2^T\mathbf{z}_2 & \cdots & \mathbf{z}_2^T\mathbf{z}_S \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{z}_S^T\mathbf{z}_1 & \mathbf{z}_S^T\mathbf{z}_2 & \cdots & \mathbf{z}_S^T\mathbf{z}_S \end{pmatrix}. \tag{20.87}$$

To accomplish the inversion of $(\mathbf{I}+\mathbf{C})$, required in (20.83), we apply the spectral theorem for Hermitian matrices [416]. The theorem allows for an orthogonal eigenvalue decomposition of $\mathbf{C}$ in the form

$$\mathbf{C} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T,$$

where $\mathbf{V}$ is a matrix whose columns are orthogonal eigenvectors, and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the eigenvalues of $\mathbf{C}$.

The final point about the MLEF is the updating of the square root analysis error covariance matrix by

$$\mathbf{P}_a^{\frac{1}{2}} = \mathbf{P}_f^{\frac{1}{2}} \left(\mathbf{I} + \mathbf{C}\left(\mathbf{x}_{opt}\right)\right)^{-\frac{T}{2}}, \tag{20.88}$$

where $\mathbf{x}_{opt}$ is approximately the minimum of (20.82). This update to the covariance matrix is similar in appearance to that in [38], but the main difference is that we have not restricted the observation operator to be linear. Another important difference is that, by using the cost function, we are able to allow for non-Gaussian errors. This is possible through the cost functions defined in [135–137]. One last comment about the difference between the MLEF and some of the other ensemble filters is that, by using (20.82), the minimization provides the iterative solution to the nonlinear analysis problem, rather than assuming a linear solution, as is the case in the Kalman filter.

An important feature to note here is that the MLEF **does not** calculate an ensemble mean at any point. Another point to observe is that the MLEF does not need any variance inflation or localization, although localization has since been added to the MLEF; see [508] for exact details.

Fletcher and Zupanski [138] show that there appears to be an underlying structure to the MLEF that the other ensemble filters do not have. In [138] they investigate the link between number of ensemble members and different dynamical flows in the Colorado State University global shallow water equation model with the Rossby-Haurwitz wave. The surprising feature that was detected was that for fast flow, we did not need as many members as for a flow that was near geostrophic balance. Upon further investigation, it was found that some of the steps in the analysis update leads to the generation of what appear to be hybrid Lyapunov and bred vectors. We shall briefly summarize these findings here.

20.5.3 Lyapunov and Bred Vectors

Bred vectors were first presented as finite amplitude, finite time extensions of Lyapunov vectors in [440,441]. A clearer mathematical description can be found in [208], where a comparison is made between bred vectors and Lyapunov vectors. The two different vectors are briefly summarized below.

Lyapunov vector generation

For Lyapunov vectors it is assumed that there is an evolving basic solution, $\mathbf{f}(\mathbf{x}, t)$, that satisfies a set of nonlinear model equations discretized in time and space. The associated numerical integration scheme is represented by

$$\mathbf{f}(\mathbf{x}, t + \Delta t) = \mathcal{M}\left(\mathbf{f}(\mathbf{x}, t)\right).$$

If the initial conditions are perturbed, then the linear evolution of that perturbation is defined by

$$\delta\mathbf{f}(\mathbf{x}, t + \Delta t) = \mathbf{L}(\mathbf{f}, t, \Delta t)\,\delta\mathbf{f}(\mathbf{x}, t), \tag{20.89}$$

where

$$\mathbf{L} = \frac{\partial \mathcal{M}}{\partial \mathbf{x}}, \tag{20.90}$$

is the tangent linear model, TLM, or propagator.

Given the descriptions in (20.89) and (20.90), the procedure to calculate the leading Lyapunov vectors starts with the arbitrary perturbation, $\delta\mathbf{f}(\mathbf{x}, t)$, and evolve this perturbation through the TLM

(20.89). After a sufficiently long time, the perturbation converges to the leading Lyapunov vector. To find the other Lyapunov vectors, some other perturbations can be used, but must be orthogonalized at each time step to prevent convergence to the leading Lyapunov vector.

Bred vector generation

The procedure to generate bred vectors starts with an arbitrary perturbation, $\delta\mathbf{f}(\mathbf{x}, t)$, of some size, defined by an arbitrary norm. The perturbation is then added to the control solution. This perturbed initial condition is integrated with the nonlinear model to the next time step, and then subtracted from the trajectory of the control run. The evolved perturbation, $\widehat{\delta\mathbf{f}}$, is given by

$$\widehat{\delta\mathbf{f}}(\mathbf{x}, t+\Delta t) = \boldsymbol{\mathcal{M}}(\mathbf{f}(\mathbf{x}, t) + \delta\mathbf{f}(\mathbf{x}, t+\Delta t)) - \boldsymbol{\mathcal{M}}(\mathbf{f}(\mathbf{x}, t)). \tag{20.91}$$

The next step is to measure the increase in the size of the evolved perturbation, and then rescale to create the perturbation for the next nonlinear run. In the process of calculating the bred vectors we do not orthogonalize the vectors if we have multiple perturbations. Therefore, all of the bred vectors are related to the leading Lyapunov vector [207].

We now consider the structure of the MLEF to compare to the techniques described above.

20.5.4 Hybrid Lyapunov-Bred Vectors

The two main matrices for the MLEF are the square root analysis error covariance matrix, $\mathbf{P}_{\boldsymbol{a}}^{\frac{1}{2}}$ (20.88) with (20.84); and the square root forecast error covariance matrix, $\mathbf{P}_{f}^{\frac{1}{2}}$ (20.81). Considering $\mathbf{P}_{f}^{\frac{1}{2}}$ first, we have

$$\mathbf{P}_{f}^{\frac{1}{2}} = \begin{pmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \dots & \mathbf{b}_S \end{pmatrix},$$
$$\mathbf{b}_j = \boldsymbol{\mathcal{M}}_{i-1,i}\left(\mathbf{x}_i + \mathbf{p}_j\right) - \boldsymbol{\mathcal{M}}_{i-1,i}(\mathbf{x}_i); \quad j = 1, 2, \dots, S,$$

where $\mathbf{p}_j$s are the columns of $\mathbf{P}_{a}^{\frac{1}{2}}$ from the previous cycle. Given the definition for bred vectors in (20.91), it is clear that the columns of $\mathbf{P}_{f}^{\frac{1}{2}}$ are a series of nonlinearly evolved perturbations. We now consider how these perturbations are calculated to see how they fit into either the rescaling criterion of the bred vectors generation, or the orthogonalization criterion of separating Lyapunov vectors, or both.

We start be recalling the definition of the Hessian update (20.88),

$$\mathbf{P}_{\boldsymbol{a}}^{\frac{1}{2}} \equiv \mathbf{P}_{f}^{\frac{1}{2}}\left(\mathbf{I} + \mathbf{C}\left(\mathbf{x}_{opt}\right)\right)^{-\frac{1}{2}}, \tag{20.92}$$

where

$$\mathbf{C} = \mathbf{P}_{\boldsymbol{f}}^{\frac{T}{2}}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{P}_{f}^{\frac{1}{2}}. \tag{20.93}$$

As we can see in (20.92), the inverse of the Hessian matrix of the cost function is required. This inversion is accomplished through an orthogonal eigenvalue decomposition. Therefore, (20.92) can be rewritten as

$$\mathbf{P}_{a}^{\frac{1}{2}} = \mathbf{P}_{f}^{\frac{1}{2}}\mathbf{V}(\mathbf{I} + \Lambda)^{-\frac{1}{2}}\mathbf{V}^T, \tag{20.94}$$

where $\mathbf{V}$ is the matrix containing the orthogonal basis of the Hessian of the cost function, projected on to the ensemble subspace.

If we now consider the sequence of matrix multiplications in (20.94), we see that we first apply an orthogonal transformation to $\mathbf{P}_f^{\frac{1}{2}}$. This transformation could be in the form of any orthogonal transformation, i.e., rotation, reflection, etc.; see [416] or [157] for more information about the different types of orthogonal transforms. To these new transformed states we rescale with respect to the inverse square root of the Hessian matrix in ensemble subspace. This tells us the directions in which the maximum energy/variability occurs and we scale the increments accordingly. We finally transform this new state again by the inverse transform matrix, i.e., $\mathbf{V}^T$.

Therefore, the reason why the filter is referred to as using a form of hybrid Lyapunov-bred vectors is because the evolution of the perturbations are through the nonlinear model, similar to bred vectors, but when we consider the Hessian update we both rescale the perturbations with respect to the directions with the most variance, i.e., the inverse eigenvalues, and apply orthogonal transforms to the perturbations, which is similar to Lyapunov vector generation.

As it appears that the filter has a Lyapunov vector generation component implicitly built in, it was noticed in [138] that the performance of the MLEF was consistent with previous studies with Lyapunov and bred vectors. As mentioned in [208], the more stable the dynamics, i.e., high Burger number, low Rossby number, or low Burger number, low Rossby number (geostrophic balance), then there are more equally dominant Lyapunov vectors. The other case study, which used fast, tall wave with more dynamics involved, showed us that we require fewer vectors/ensemble members, which is again consistent with the behavior of Lyapunov vectors, as seen in previous studies [271].

A technique was developed for the MLEF, and effectively the ETKF, that allowed for the inflation of the size of the ensemble through using the perturbations at the previous analysis cycle, augmented to the current ensemble. This introduces temporal information into the ensemble, as well as information about the dominant directions at the previous analysis time [445]. Also in [445], a technique is developed to reduce the size of the ensemble when there are ensemble members whose contribution to the total variance is nearly zero. This is consistent with ensemble members pointing in the same Lyapunov direction, therefore, not adding much to the analysis.

It was shown in [138] that with larger ensembles, there was the possibility that some of the members might tend to the same Lyapunov vectors, and hence cause a form of degeneracy in the $\mathbf{C}$ matrix. This would imply that this could be an ill-conditioned problem, but still be symmetric positive definite $\mathbf{C}$ matrix. If an eigenvalue were to be zero, then there would be two ensemble members pointing in the same direction. This could lead to problems with the inversion of the $\mathbf{C}$ matrix. The new technique in [445] enables the removal of these members, thus resizing and reconditioning the $\mathbf{C}$ matrix to remove degeneracy.

20.5.5 MLEF, Information Theory, and Entropy Reduction

In [504] it is proposed to use the formulation of the MLEF to apply information theory in ensemble space. The starting point is the degree of freedom for signal or DOF for signal, denoted d_s and is commonly defined in terms of analysis and forecast error covariances, $\mathbf{P}_a$ and $\mathbf{P}_f$ as before. Thus we have

$$d_s = tr\left[\mathbf{I}_N - \mathbf{P}_a\mathbf{P}_f^{-1}\right], \tag{20.95}$$

where tr denotes the trace operation. The quantity d_s counts the number of new pieces of information brought to the analysis by the observations, with respect to what is already known, as expressed by $\mathbf{P}_f$. As d_s is dependent on the ratio between the analysis and forecast error covariance matrices, $\left(\mathbf{P}_a\mathbf{P}_f^{-1}\right)$, d_s measures the forecast error reduction due to the new information from the observations. In [458] d_s is defined in terms of a so-called influence matrix, $\mathbf{A}$, as

$$d_s = tr\left[\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}_a\mathbf{H}^T\mathbf{R}^{-\frac{1}{2}}\right] = tr\left[\mathbf{A}\right], \tag{20.96}$$

and in [122] it is shown that (20.95) and (20.96) are equivalent. Now using the definition from earlier in this section for $\mathbf{P}_a^{\frac{1}{2}} = \mathbf{P}_f^{\frac{1}{2}}\left(\mathbf{I}+\mathbf{C}\right)^{-\frac{1}{2}}$, and using the property $tr\left(\mathbf{x}\mathbf{x}^T\right) = tr\left(\mathbf{x}^T\mathbf{x}\right)$, it is possible to write (20.96) in ensemble space as

$$d_s = tr\left[\left(\mathbf{I}+\mathbf{C}\right)^{-1}\mathbf{P}^{\frac{T}{2}}\mathbf{H}^T\mathbf{R}^{-\frac{T}{2}}\mathbf{R}^{-\frac{1}{2}}\mathbf{H}\mathbf{P}_f^{\frac{1}{2}}\right]. \tag{20.97}$$

The next step is to recognize that the sets of matrices are what was defined as $\mathbf{Z}$ earlier, which implies that we can write d_s in ensemble space as

$$d_s = tr\left[\left(\mathbf{I}+\mathbf{C}\right)^{-1}\mathbf{Z}^T\mathbf{Z}\right] = tr\left[\left(\mathbf{I}+\mathbf{C}\right)^{-1}\mathbf{C}\right]. \tag{20.98}$$

The introduction of the $\mathbf{C}$ matrix defined a link between information theory and ensemble data assimilation. When calculating information content measures such as d_s, a flow-dependent $\mathbf{P}_f$ obtained directly form ensemble data assimilation is used.

Once the information matrix $\mathbf{C}$ is known, it is possible to define various information measures, and is especially useful to define these measure in terms of the eigenvalues, λ_i^2, of $\mathbf{C}$. Two measures that are presented are d_s and entrophy reduction h as

$$d_s = \sum_i \frac{\lambda_i^2}{1+\lambda_i^2}, \quad h = \frac{1}{2}\sum_i \ln\left(1+\lambda_i^2\right). \tag{20.99}$$

The reader is referred to [504] for results involving the RAMS model.

An important feature to notice about the MLEF compared to the other ensemble approaches is that it finds a minimum of a maximum likelihood cost function similar to 3D VAR, but here we use the fact that it is possible to project the solver on to the ensemble space. In 2008 the MLEF was re-designated not as an ensemble KF-based method but as a **hybrid method**. The MLEF is not a true hybrid system; the first reason is that it does not use any static covariances, and secondly it does not update the variance through the mean, which is why more recently the matrices involved with the MLEF are referred to as **uncertainty matrices** rather than as covariance matrices. To illustrate that the MLEF is not a full hybrid method, we shall now introduce the different types of hybrid data assimilation that have been proposed and are now in operations at many different numerical weather and ocean prediction centers.

20.6 Hybrid Ensemble and Variational Data Assimilation Methods

There are many different forms of hybrid data assimilation schemes, and we shall only present a few of the more frequently used versions. We should note that new ways to combine different strengths of both the ensemble methods include variational, analysis correction, and OI.

We saw a somewhat hybrid system in the last section with the MLEF that minimizes for the mode of the posterior distribution in ensemble space but evolves the forecast/background error covariance matrix so that they are 100% flow dependent.

The first real mention of a hybrid system, where an EnKF is combined with a 3D VAR analysis scheme, is in Hamil and Snyder [164]. Their hybrid approach is to take the standard 3D VAR cost function with a linear observation operator

$$J(\mathbf{x}) = \frac{1}{2}\left(\mathbf{x} - \mathbf{x}x^b\right)^T \mathbf{B}^{-1}\left(\mathbf{x} - \mathbf{x}^b\right) + \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{x})^t \mathbf{R}^{-1}(\mathbf{y} - \mathbf{H}\mathbf{x}), \tag{20.100}$$

where the background error covariance matrix $\mathbf{B}$ is decomposed in the form $\mathbf{B} = \mathbf{SCS}^T$, where $\mathbf{S}$ is the transform from spectral coefficients to grid points and $\mathbf{C}$ is the diagonal matrix of variances of the spectral coefficients and seek the analysis increments, $\mathbf{x}^a - \mathbf{x}^b$, such that

$$\left(\mathbf{I} + \mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\right)\left(\mathbf{x}^a - \mathbf{x}^b\right) = \mathbf{B}\mathbf{H}^T\mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{H}\mathbf{x}^\mathbf{b}\right). \tag{20.101}$$

The hybrid EnKF-3D VAR scheme uses a weighted mean of $\mathbf{B} = \mathbf{SCS}^T$ and $\mathbf{P}^b$ derived from the ensemble, which is fully time dependent and spatially inhomogeneous, such that

$$\mathbf{B} = (1-\alpha)\mathbf{P}^b + \alpha\mathbf{SCS}^T. \tag{20.102}$$

By changing the value of α from 0 to 1, the analysis changes from using only flow-dependent ensemble-based error covariances to using the original 3D VAR static covariances.

The advantages of the hybrid approach as set out in [164] are:

1. The hybrid scheme allows the user to evaluate combinations of 3D VAR and ensemble-based background statistics rather than relying strictly upon one or the other.
2. Ensemble-based statistics alone will be rank deficient and subject to sampling errors, and as such blending in the 3D VAR static statistics may *fill out* the covariance matrix and ameliorate some of the sampling error problems.

To implement the hybrid scheme requires an appropriate ensemble and in [164] the authors state that each ensemble member is used in turn as the background state in (20.101), where $\mathbf{B}$ is given by (20.102) and is updated upon distinct perturbed observations. To update the $\mathbf{P}^b$ matrix, Hamil and Snyder employ an approach where the ith member from the sample excludes that member from the calculation, where mathematically this is

$$\mathbf{P}_i^b = \frac{1}{n-2}\sum_{j=1, j\neq i}^{n}\left(\mathbf{x}_j^b - \overline{\mathbf{x}}_i^b\right)\left(\mathbf{x}_j^b - \overline{\mathbf{x}}_i^b\right)^T, \tag{20.103}$$

where the mean in (20.103) has also been calculated excluding the ith member.

20.6.1 α Control Variables

The α control variables were introduced at the United Kingdom's Met. Office in the late 1990s and the theory behind them was mentioned in Lorenc [261]. This technique has been implemented operationally at the Met. Office as well as in the National Center for Environmental Prediction (NCEP) initial hybrid system [465]. We shall consider the approach set out in Clayton et al. [67]. The motivation of the hybrid approach at the Met. Office was to capture the *errors of the day* which are the short-range errors, while the climatological or static covariance matrix of the variational component captures the large- and longer-scale errors.

For the ensemble component of the hybrid system at the Met. Office they run the Met. Office Global and Regional Ensemble Prediction Systems, or MOGREPS, which comprises of an ETKF as described in Wang et al. [464]. The global version of the ensemble prediction system is called MOGREPS-G.

An important feature of the hybrid system at the Met. Office is that it is completely coupled. This coupling comes from:

1. the analysis perturbations for the ensemble system are generated by centering around the deterministic analysis from the 4D VAR system; and
2. the 4D VAR system is dependent on forecast data from the ensemble system.

At the start of each 4D VAR window the necessary ensemble forecast fields are taken from MOGREPS-G and interpolated to the analysis grid of the 4D VAR system. Note that the ensemble members are ran at a lower resolution to both the analysis scheme in the 4D VAR and the global forecast model. As we have seen with the other ensemble filters, the differences between the ensemble members fields and the ensemble mean are taken, scaled by $\frac{1}{\sqrt{K-1}}$, and stored in an array $\mathbf{W}$ such that

$$\mathbf{P}_e^f = \mathbf{W}\mathbf{W}^T. \tag{20.104}$$

As with most ensemble-based approximations to the error covariance matrix, (20.104) is undersampling the errors, and so the Met. Office applies a localization matrix, $\mathbf{C}$, such that

$$\mathbf{B}_e = \mathbf{P}_e^f \circ \mathbf{C}. \tag{20.105}$$

Therefore, given the static, climatological background error covariance matrix from the 4D VAR system, the hybrid system seeks to implement a hybrid background error covariance matrix that is a linear combination of the static and flow dependent background error covariance matrices as

$$\mathbf{B} = \beta_c^2 \mathbf{B}_c + \beta_e^2 \mathbf{B}_e, \tag{20.106}$$

where β_c^2 and β_e^2 are scalar weights. This is where the α control variables come in, or implementing an extended control variable method as it is referred to as in [67].

The process of implementing the α control variable approach starts with the decomposition that the Met. Office implements for its control variable transform. The Met. Office uses an incremental 4D VAR system that involves increments, $\delta\mathbf{w}$, and a control variable, $\mathbf{v}$, such that $\delta\mathbf{w} = \mathbf{U}\mathbf{v}$, where $\mathbf{U}$ is the square root matrix of $\mathbf{B}_c$, which are the inverses of the transforms $\mathbf{T}$ that make the control variables uncorrelated with approximately unit variance. The transforms associated with the $\mathbf{T}$ matrix are referred to as the T-Transforms and comprise of

$$\mathbf{T} = \mathbf{T}_h \mathbf{T}_v \mathbf{T}_p, \tag{20.107}$$

where $\mathbf{T}_p$ combines some fields to reduce the total number of fields; $\mathbf{T}_v$ is the projection on to approximately uncorrelated vertical modes; and $\mathbf{T}_h$ is the projection on to global spherical harmonic functions.

The $\mathbf{U}$ transform is made up of the approximate, or exact, inverses of the $\mathbf{T}$ transforms, such that

$$\mathbf{U} = \mathbf{U}_p \mathbf{U}_v \mathbf{U}_h. \tag{20.108}$$

Thus the increment $\delta\mathbf{w}$ is related to the static and ensemble components through

$$\delta\mathbf{w} = \beta_c \mathbf{U}_p \mathbf{U}_v \mathbf{U}_h \mathbf{v} + \beta_e \sum_{k=1}^{K} \mathbf{w'}_k \circ \boldsymbol{\alpha}_k, \tag{20.109}$$

where $\mathbf{w'}_k$ is the ensemble error mode.

Given the definition in (20.109), the incremental 4D VAR cost function becomes

$$J(\mathbf{v}, \boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_k) = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \sum_{k=1}^{K} \boldsymbol{\alpha}_k^T \mathbf{C}\boldsymbol{\alpha}_k + J_o + J_c, \tag{20.110}$$

where the J_0 term is the observational component of the cost function, and J_c is any constraint that we wish to enforce on the minimum of the cost function. It is possible to improve the conditioning of the minimization of the cost function in (20.110) by introducing an α control vector $\mathbf{v}^\alpha$ which is the concatenation of the K vectors $\mathbf{v}_k^\alpha$ such that

$$\boldsymbol{\alpha}_k = \mathbf{U}^\alpha \mathbf{v}_k^\alpha, \tag{20.111}$$

where $\mathbf{U}^\alpha \equiv \mathbf{C}^{\frac{1}{2}}$.

Therefore, the cost function in (20.110) now becomes

$$J(\mathbf{v}, \mathbf{v}^\alpha) = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{v}^\alpha)^T(\mathbf{v}^\alpha) + J_0 + J_c. \tag{20.112}$$

The matrix $\mathbf{U}$ is a localization matrix in both the horizontal and vertical; more exact details can be found in [67]. The Met. Office also employs a balance preserving localization.

This type of hybrid ensemble-variational scheme is referred to as **En4D VAR**. There is of course a three-dimensional version which is referred to as **En3D VAR**, which we present now.

En3D VAR

A more detailed description of a similar scheme to that at the Met. Office, but in 3D for the NCEP hybrid system, can be found in [462,465]; there are subtle difference between the two approaches. The first feature that is different is the vertical control variable decomposition, as NCEP uses the recursive filter [168]. Another different feature is the inflation factors that are used for the different hybrid system; in the NCEP system the inflation factors are both multiplicative and additive, while the MOGREPS-G system uses an approach described in Flowerdew and Bowler [140], for the horizontal inflation and in Flowerdew and Bowler [141] for the vertical inflation. These inflation factors are based upon using radiosondes and select satellite sounding data to provide horizontally and vertically local calibration of perturbation magnitudes to root mean error in the United Kingdom's Met. Office global ensemble system. While the inflation applied in [462,465] is based on the multiplicative-additive version set out

in Whitaker and Hamill [474], their approach is different to the multiplicative-additive approach we introduced earlier.

The inflation scheme introduced in [474] is called the **relaxation to prior spread** and is a purely multiplicative inflation as it relaxes the ensemble standard deviation back to the prior through

$$\sigma^a \leftarrow (1-\alpha)\,\sigma^a + \sigma^b, \tag{20.113}$$

where $\sigma^b = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} x_i'^{b^2}}$ and $\sigma^a = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} x_i'^{a^2}}$ are the prior and posterior ensemble standard deviations at each analysis grid point, and n is the ensemble size.

In [462] the authors present the solution method in quite a similar way to [67] but show how the extended control variables are found. Given the incremental 3D VAR cost function

$$J(\delta\mathbf{x}) = \frac{1}{2}(\delta\mathbf{x})^T\,\mathbf{B}_c^{-1}\delta\mathbf{x} + \frac{1}{2}(\mathbf{d}-\mathbf{H}\delta\mathbf{x})^T\,\mathbf{R}^{-1}(\mathbf{d}-\mathbf{H}\delta\mathbf{x}), \tag{20.114}$$

where $\mathbf{d}$ is the nonlinear observation innovation vector, we define the hybrid increment as

$$\delta\widehat{\mathbf{x}} = \delta\mathbf{x} + \sum_{k=1}^{K}\boldsymbol{\alpha}_k \circ \delta\mathbf{x}_k^e. \tag{20.115}$$

The associated hybrid 3D VAR cost function is given by

$$\begin{aligned} J(\delta\mathbf{x},\boldsymbol{\alpha}) &= \beta_1 J_b + \beta_2 J_e + J_o, \\ &= \frac{\beta_1}{2}(\delta\mathbf{x})^T\,\mathbf{B}_c^{-1}\delta\mathbf{x} + \frac{\beta_2}{2}\boldsymbol{\alpha}^T\mathbf{A}^{-1}\boldsymbol{\alpha} + \frac{1}{2}(\mathbf{d}-\mathbf{H}\delta\mathbf{x})^T\,\mathbf{R}^{-1}(\mathbf{d}-\mathbf{H}\delta\mathbf{x}), \end{aligned} \tag{20.116}$$

where $\mathbf{A}$ is a block diagonal matrix that defines the spatial correlation of $\boldsymbol{\alpha}$. An important feature to note from [462] is that the second term in (20.116) is preconditioned on $\mathbf{A}$, not the ensemble covariance.

As with the Met. Office system, NCEP apply a control variable transform. Following the same setup for the α control variables as described above, but where the localization and inflation terms are different, we define a new control variable

$$\delta\widetilde{\mathbf{x}} = \begin{pmatrix} \delta\mathbf{x} \\ \boldsymbol{\alpha} \end{pmatrix}. \tag{20.117}$$

This enables us to write the hybrid increment in (20.115) as

$$\delta\widehat{\mathbf{x}} = \delta\mathbf{x} + \mathbf{D}\boldsymbol{\alpha} = \begin{pmatrix} \mathbf{I} & \mathbf{D} \end{pmatrix}\begin{pmatrix} \delta\mathbf{x} \\ \boldsymbol{\alpha} \end{pmatrix} = \mathbf{C}\delta\widetilde{\mathbf{x}}. \tag{20.118}$$

We now denote the hybrid background error covariance matrix as

$$\mathbf{B} = \begin{pmatrix} \frac{1}{\beta_1}\mathbf{B}_c & \mathbf{0} \\ \mathbf{0} & \frac{1}{\beta_2}\mathbf{A} \end{pmatrix}. \tag{20.119}$$

Taking the gradients of (20.116) with respect to $\delta\mathbf{x}$ and $\boldsymbol{\alpha}$ results in

$$\nabla_{\delta\mathbf{x}} J = \beta_1\mathbf{B}_\mathbf{c}^{-1}\delta\mathbf{x} + \mathbf{H}^\mathbf{T}\mathbf{R}^{-1}(\mathbf{H}\delta\mathbf{x}-\mathbf{d}), \tag{20.120a}$$

$$\nabla_{\boldsymbol{\alpha}} J = \beta_2\mathbf{A}^{-1}\boldsymbol{\alpha} + \mathbf{D}^T\mathbf{H}^\mathbf{T}\mathbf{R}^{-1}(\mathbf{HC}\delta\mathbf{x}-\mathbf{d}). \tag{20.120b}$$

Therefore, we can also apply a preconditioner, as described in [462], so that it is possible to use a preconditioned conjugate gradient solver to the two gradients above.

The two forms of hybrid approaches that we have shown here have been for the model space-based variational systems. We now consider the PSAS-based hybrid system that has been developed by the US Navy for their numerical weather prediction system.

20.6.2 Hybrid Ensemble Transform PSAS

In this section we shall only briefly summarize the work from Mclay et al. [289,290], which takes the formulation of the ETKF but instead of updating through the Kalman filter equations, they take a best estimate of the error covariance matrix, written in the form

$$\mathbf{P}_{\mathbf{g}}^{\mathbf{a}} = \mathbf{F}\mathbf{D}_{\mathbf{g}}^{\mathbf{a}}\mathbf{F}^{\mathbf{T}}, \tag{20.121}$$

where $\mathbf{F} = [\mathbf{1}, \mathbf{2}, \ldots, \mathbf{N}]$ is a $N \times N$ orthogonal matrix that containing the eigenvectors of $\mathbf{P}_{\mathbf{g}}^{\mathbf{a}}$, and $\mathbf{D}_g^a$ is the diagonal matrix of the associated eigenvalues. Next let $\delta\mathbf{X}^{\mathbf{f}} = \left[\delta\mathbf{x}_{\mathbf{1}}^{\mathbf{f}}, \delta\mathbf{x}_{\mathbf{2}}^{\mathbf{f}}, \ldots, \delta\mathbf{x}_{\mathbf{N}}^{\mathbf{f}}\right]$ be a $N \times K$ matrix whose ith column is given by

$$\delta\mathbf{x}_i^f = \mathbf{x}_i^f - \overline{\mathbf{x}}_i^f \quad \text{where } \overline{\mathbf{x}}_i^f \equiv \frac{1}{K}\sum_{i=1}^{K}\mathbf{x}_i^f, \tag{20.122}$$

where $\mathbf{x}_i^f$ is the ith ensemble member's forecast and the overbar represents the ensemble mean. An assumption made in [289] is that the increments are approximately in dynamical balance, as to be free of spurious gravity waves.

The ensemble transform, ET hereafter, generates K initial perturbations $\delta\mathbf{X}^{\mathbf{a}} = \left[\delta\mathbf{x}_{\mathbf{1}}^{\mathbf{a}}, \delta\mathbf{x}_{\mathbf{2}}^{\mathbf{a}}, \ldots, \delta\mathbf{x}_{\mathbf{K}}^{\mathbf{a}}\right]$ from K forecast perturbations via a $K \times K$ transformation matrix $\mathbf{T} = [\mathbf{t}_{\mathbf{1}}, \mathbf{t}_{\mathbf{2}}, \ldots, \mathbf{t}_{\mathbf{K}}]$ using the formula

$$\delta\mathbf{X}^{\mathbf{a}} = \delta\mathbf{X}^{\mathbf{f}}\mathbf{T} \quad \text{where } \mathbf{T} = \mathbf{C}\,(\mathbf{\Gamma})^{-\frac{\mathbf{1}}{\mathbf{2}}}\,\mathbf{C}^{\mathbf{T}}, \tag{20.123}$$

where $\mathbf{\Gamma} = \text{diag}\,(\gamma_{11}, \gamma_{22}, \ldots, \gamma_{KK})$,

$$\mathbf{\Gamma} \equiv \begin{pmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & 1 \end{pmatrix}, \tag{20.124}$$

$\mathbf{G}$ is the diagonal matrix that contains the non-zero eigenvalues of

$$\frac{1}{N}\delta\mathbf{X}^{\mathbf{f}^{\mathbf{T}}}\left(\mathbf{P}_{\mathbf{g}}^{\mathbf{a}}\right)^{-\mathbf{1}}\delta\mathbf{X}^{\mathbf{f}} = \mathbf{C}\mathbf{\Lambda}\mathbf{C}^{\mathbf{T}}, \tag{20.125}$$

where $\mathbf{C}$ contains the orthogonal eigenvectors of the symmetric matrix on the left-hand side of (20.125). The reason why one is in the matrix in (20.124) is to replace the zero eigenvalue that occurs in the decomposition above due to requirement that the sum of the forecast perturbations equals zero.

As a result of this approach, it now means that the analysis increments used to initialize the ensemble members are orthogonal.

To exploit this orthogonality condition of the analysis perturbation, the authors employ ideas similar to the LETKF, where they partition the globe into different regions so that the block, or banded ET forecast ensembles, perform better than the global ET forecasts.

Now, moving on to the hybrid system in observation space, as set out in [226], we have that they **do not** use the extended control variable approach, but instead use the approach from [164]. The ensemble component of the hybrid system in NAVDAS-AR is obtained through the process described in [289, 290], but now the $\mathbf{P}_{\mathbf{0}}^{\mathbf{b}}$ term in the representer approach used in NAVDAS-AR is defined as

$$\mathbf{P}_{\mathbf{0}}^{\mathbf{b}} = (\mathbf{1} - \alpha)\,\mathbf{P}_{\mathbf{static}}^{\mathbf{b}} + \alpha \mathbf{P}_{\mathbf{flow}}^{\mathbf{b}}. \tag{20.126}$$

An important feature to notice here is that at the time of writing, the humidity component of the static error covariance matrix is univariate, while it is multivariate in the flow-dependent error covariance matrix. As with all of the ensemble-based systems, a form of localization is performed but in model space as work done by Campbell et al. [54], showed that localizing in observation space degrades the assimilation of radiances, which make up a large proportion of the observations available to most operational numerical weather prediction data assimilation schemes [54].

20.6.3 Ensembles of 4D VARs (EDA)

We now move on to the last of the hybrid-based systems, which is referred to as the **ensemble of 4D VARs** or **EDA**. This approach involves running an ensemble of independent lower-resolution 4D VAR assimilations that can differ by perturbing observations, boundary conditions, and or model physics [194]. Given this ensemble of analyses, we average over them to form a flow-dependent estimate for the background error covariance matrix that is then combined with the static matrix to produce the operational, or control, analysis and forecast. This approach is used operationally at ECMWF and at Mètèo-France at different resolutions and with a different number of VAR systems. A detailed description of the evolution of the ensemble of 4D VARs at ECMWF can be found in [44].

We now move on to a new version of data assimilation that is being developed in the atmospheric field which is referred to as **NDEnVAR**.

20.7 NDEnVAR

According to Lorenc [257], the data assimilation systems that have the designate NDEnVAR, where the ND stands for number of dimensions, i.e., 3D or 4D, are variational-based data assimilation systems that only use ensemble covariances, but in the four-dimensional case they do not use the adjoint of the tangent linear models.

A clear mathematical and algorithmic description of the NDEnVAR systems can be found in Desroziers et al. [96], but we shall present the brief description from Lorenc et al. [264].

For this section we shall consider four-dimensional trajectories, which can be seen as a sequence of three-dimensional states describing the evolution over a time window. A standard that has been introduced to represent these 4D fields in the atmospheric community is to denote the vector with an underline. Therefore let $\underline{\mathbf{x}}^b$ be the background trajectory. We then have the expected error covariance of $\underline{\mathbf{x}}^b$ as $\underline{\mathbf{P}}$. This defines a Gaussian PDF for the 4D increment $\delta\underline{\mathbf{x}}$:

$$\delta\underline{\mathbf{x}} \sim G\left(\mathbf{0}, \underline{\mathbf{P}}\right), \tag{20.127}$$

which gives the probability that $\underline{\mathbf{x}}^b + \delta\underline{\mathbf{x}}$ is the true trajectory. Thus, if we follow the usual assumptions for incremental 4D VAR, then we have to find the minimum of the following cost function:

$$J\left(\delta\underline{\mathbf{x}}\right) = \frac{1}{2}\delta\underline{\mathbf{x}}^T\underline{\mathbf{P}}^{-1}\delta\underline{\mathbf{x}} + \frac{1}{2}\left(\underline{\mathbf{y}} - \underline{\mathbf{y}}^o\right)^T\underline{\mathbf{R}}^{-1}\left(\underline{\mathbf{y}} - \underline{\mathbf{y}}^o\right), \tag{20.128}$$

where for the observational component in (20.128), $\underline{\mathbf{y}}^0$ is the observations in the time window, and $\underline{\mathbf{y}}$ is the nonlinear observation operator acting in time given by

$$\underline{\mathbf{y}} - \underline{\mathbf{H}}\left(\underline{\mathbf{x}}^b + \delta\underline{\mathbf{x}}\right), \tag{20.129}$$

where $\underline{\mathbf{H}}$ includes time, horizontal, and vertical interpolation, and potentially nonlinear calculations.

Incremental 4D VAR system replace the true state, $\mathbf{x}^t$, by $\mathbf{x}^b + \delta\mathbf{x}$ along with the assumption;

$$\underline{\mathcal{M}}\left(\mathbf{x}^b + \delta\mathbf{x}\right) = \underline{\mathcal{M}}\left(\mathbf{x}^b\right) + \delta\underline{\mathbf{x}}, \tag{20.130a}$$

$$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\delta\mathbf{x}. \tag{20.130b}$$

Applying a reduction of the control variable technique means that the 4D covariance matrix, which is implicit inside of 4D VAR, is given by

$$\underline{\mathbf{P}} = \underline{\mathbf{M}}\mathbf{P}\underline{\mathbf{M}}^{\mathbf{T}}, \tag{20.131}$$

where $\mathbf{P}$ is the 3D error covariance matrix at the beginning of the window. If we apply the $\mathbf{T}$ and $\mathbf{U}$ transforms as we showed earlier in the summary of the Met. Office's En4D VAR system, it is possible to rewrite the cost function in terms of the control variables $\mathbf{v}$ such that $\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\mathbf{U}\mathbf{v}$,

$$J\left(\mathbf{v}\right) = \frac{1}{2}\mathbf{v}\mathbf{v}^{\mathbf{T}} + \frac{\mathbf{1}}{\mathbf{2}}\left(\underline{\mathbf{y}} - \underline{\mathbf{y}}^{\mathbf{o}}\right)\underline{\mathbf{R}}^{-\mathbf{1}}\left(\underline{\mathbf{y}} - \underline{\mathbf{y}}^{\mathbf{o}}\right). \tag{20.132}$$

The associated 4D error covariance matrix is given by

$$\underline{\mathbf{P}} = \underline{\mathbf{M}}\mathbf{U}\mathbf{U}^{\mathbf{T}}\underline{\mathbf{M}}^{\mathbf{T}}. \tag{20.133}$$

Ensemble 4D VAR implicitly uses localized ensemble error instead of the climatological covariance $\mathbf{B}$:

$$\mathbf{P} = \mathbf{C} \circ \mathbf{X}\mathbf{X}^{\mathbf{T}}, \tag{20.134}$$

where $\mathbf{X} = \left[\mathbf{x}'_{\mathbf{1}}, \mathbf{x}'_{\mathbf{2}}, \ldots, \mathbf{x}'_{\mathbf{N}}\right]$ is the array of normalized ensemble perturbations valid at the beginning of the window. We apply the α control variable technique

$$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\sum_k \boldsymbol{\alpha} \circ \mathbf{x}'_k. \tag{20.135}$$

Each $\boldsymbol{\alpha}_k$ is smooth using the technique $\boldsymbol{\alpha}_k = \mathbf{U}^{\alpha}\mathbf{v}^{\alpha}_{\mathbf{k}}$, such that

$$\mathbf{C} = \mathbf{U}^{\alpha}\left(\mathbf{U}^{\alpha}\right)^{\mathbf{T}}. \tag{20.136}$$

All of the $\mathbf{v}_\mathbf{k}^\alpha$ are concatenated into a single vector $\mathbf{v}$ so that the background component of the cost function is transformed into

$$\frac{1}{2}\mathbf{v}\mathbf{v}^\mathbf{T} = \frac{\mathbf{1}}{\mathbf{2}}\sum_\mathbf{k}\mathbf{v}_\mathbf{k}^\alpha\left(\mathbf{v}_\mathbf{k}^\alpha\right)^\mathbf{T}. \tag{20.137}$$

The important feature to note here is that the error covariance matrix now becomes

$$\underline{\mathbf{P}} = \underline{\mathbf{M}}\left(\mathbf{C}\circ\mathbf{X}\mathbf{X}^\mathbf{T}\right)\underline{\mathbf{M}}^T. \tag{20.138}$$

Hybrid 4D VAR is constructed through a weighted average combining the traditional 4D VAR with the ensemble 4D VAR above, such that

$$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\left(\beta_c\mathbf{U}\mathbf{v} + \beta_\mathbf{e}\sum_\mathbf{k}\mathbf{U}^\alpha\mathbf{v}_\mathbf{k}^\alpha\circ\mathbf{x}_\mathbf{k}'\right). \tag{20.139}$$

Therefore, the error covariance matrix for the hybrid 4D VAR is given by

$$\underline{\mathbf{P}} = \underline{\mathbf{M}}\left(\beta_c^2\mathbf{B} + \beta_\mathbf{e}^\mathbf{2}\mathbf{C}\circ\mathbf{X}\mathbf{X}^\mathbf{T}\right)\underline{\mathbf{M}}^T. \tag{20.140}$$

We should note here that the square of the hybrid weights are in the definition above, and assuming that they both give independent valid estimates of $\mathbf{P}$, then we obtain the condition that $\beta_c^2 + \beta_e^2 = 1$.

4DEnVAR

This approach is similar to that of the En4D VAR but is designed to avoid the use of the tangent linear and adjoint models. The 4DEnVAR approach extends to four dimensions for all the En4D VAR equations, but now applying them to the sequence of states in trajectories rather than at the single time at the beginning of the window. This implies that we can use an implicit localized ensemble covariance:

$$\underline{\mathbf{P}} = \underline{\mathbf{C}}\circ\underline{\mathbf{X}}\underline{\mathbf{X}}^T, \tag{20.141}$$

where $\underline{\mathbf{X}} = \left[\underline{\mathbf{x}}_1', \underline{\mathbf{x}}_2', \ldots, \underline{\mathbf{x}}_N'\right]$ contains the normalized ensemble perturbations valid through the window as $\underline{\mathbf{X}} = \frac{\underline{\mathbf{x}}_k - \overline{\underline{\mathbf{x}}}}{\sqrt{N-1}}$. This method uses the α control variables but now in the form of $\underline{\boldsymbol{\alpha}}_k$, which define the local weight given to each perturbation trajectory, so that

$$\delta\underline{\mathbf{x}} = \sum_k\underline{\boldsymbol{\alpha}}_k\circ\underline{\mathbf{x}}_k'. \tag{20.142}$$

Since each $\underline{\mathbf{x}}_k'$ is a normalized difference of nonlinear forecasts, then (20.142) is a linear combination of nonlinear forecasts, using localized weights. It is a linear function of the $\underline{\boldsymbol{\alpha}}_k$s and hence is a different kind of linear model to that in (20.130b).

In its initial implementation of 4DEnVAR, the Met. Office does not allow for the variation in time of the $\underline{\boldsymbol{\alpha}}_k$s, but instead uses a persistence forecast $\underline{\mathbf{I}}$ such that $\underline{\boldsymbol{\alpha}}_k = \underline{\mathbf{I}}\underline{\boldsymbol{\alpha}}_k$. A smoothing technique is still applied to the α control variables. Therefore the 4D incremental trajectory is given by

$$\delta\underline{\mathbf{x}} = \beta_c\underline{\mathbf{I}}\mathbf{U}\underline{\mathbf{x}}\mathbf{v}^\mathbf{c} + \beta_\mathbf{e}\sum_\mathbf{k}\underline{\mathbf{I}}\mathbf{U}^\alpha\mathbf{v}_\mathbf{k}^\alpha\circ\underline{\mathbf{x}}_\mathbf{k}', \tag{20.143}$$

Table 20.1 Summary of the Increments and the Analysis Covariance Matrices for Each Data Assimilation Scheme.

Scheme	Increment	Analysis Covariance Matrix
4D VAR	$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\delta\mathbf{x}$	$\underline{\mathbf{P}} = \underline{\mathbf{M}}\mathbf{P}\underline{\mathbf{M}}^{\mathbf{T}}$
EnVAR	$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\sum_k \boldsymbol{\alpha}_k \circ \mathbf{x}'_k$	$\underline{\mathbf{P}} = \underline{\mathbf{M}}\left(\mathbf{C} \circ \mathbf{X}\mathbf{X}^{\mathbf{T}}\right)\underline{\mathbf{M}}$
En4D VAR	$\delta\underline{\mathbf{x}} = \underline{\mathbf{M}}\left(\beta_c\mathbf{U}\mathbf{v} + \beta_{\mathbf{e}}\sum_{\mathbf{k}} \mathbf{U}^{\alpha}\mathbf{v}^{\alpha}_{\mathbf{k}} \circ \mathbf{x}'_{\mathbf{k}}\right)$	$\underline{\mathbf{P}} = \underline{\mathbf{M}}\left(\beta_c^2\mathbf{B} + \beta_{\mathbf{e}}^{\mathbf{2}}\mathbf{C} \circ \mathbf{X}\mathbf{X}^{\mathbf{T}}\right)\underline{\mathbf{M}}^{T}$
4DEnVAR	$\delta\underline{\mathbf{x}} = \beta_c\underline{\mathbf{I}}\mathbf{U}\mathbf{v}^{\mathbf{c}} + \beta_{\mathbf{e}}\sum_{\mathbf{k}} \underline{\mathbf{I}}\mathbf{U}^{\alpha} \circ \underline{\mathbf{x}}'_{\mathbf{k}}$	$\underline{\mathbf{P}} = \beta_c^2\underline{\mathbf{I}}\mathbf{B}\underline{\mathbf{I}}^{\mathbf{T}} + \beta_{\mathbf{e}}^{\mathbf{2}}\underline{\mathbf{C}} \circ \underline{\mathbf{X}}\underline{\mathbf{X}}^{\mathbf{T}}$

and the implicit covariance still contains neither the square of the weights and is given by

$$\underline{\mathbf{P}} = \beta_c^2\underline{\mathbf{I}}\mathbf{B}\underline{\mathbf{I}}^{\mathbf{T}} + \beta_{\mathbf{e}}^{\mathbf{2}}\underline{\mathbf{C}} \circ \underline{\mathbf{X}}\underline{\mathbf{X}}^{\mathbf{T}}, \tag{20.144}$$

where we can see that the expression in (20.144) contains neither the tangent linear model, nor the adjoint.

While this appears to be a great alternative to 4D VAR, most results have shown that while 4DEnVAR beats the static 4D VAR and the hybrid 3D VAR, it does not appear, at the time of writing, to be able to beat En4D VAR. One possible reason, as mentioned in [264], is the fact that the 4DEnVAR does not create very well-balanced increments, but another reason could be the fact that persistence of the static component through the window is not a good choice as it is hard to believe that the large-scale motions would have the same error covariances throughout the length of the window, given the fact that incremental 4D VAR evolves the static component of the background error covariance matrix throughout the window. We know that even the static 4D VAR evolves the static covariance matrix through the Hessian of the cost function. There is a detailed study performed with the NCEP Global Forecasting System with different versions of En3D VAR, En4D VAR, 3DEnVAR, and 4DEnVAR in Kleist and Ide [217,218], and we recommend the reader to these papers for a more detailed explanation of the performances that they discovered of these different configurations of the hybrid systems.

We have compiled all the different equations for the increments and the analysis covariance matrices from this section in Table 20.1.

Just after the first edition of this textbook came out there was an extensive comparison of the hybrid variational methods for global numerical weather prediction that was a follow on to the summary just provided and the readers are referred to [266] for this study.

20.8 Scale Dependent Background Error Covariance Localization

In this section we introduce a form of localizations that was just starting to gain traction when the first edition of the textbook came out and is now used in many ensemble systems. This form of localization has been developed over a series of papers, [48,50,56]. We shall focus on the latter two but the reader is referred to [48] for the insight that started this work.

It is stated at the start of [50] that the approach presented here is similar to that from [48] but the main difference is, in this approach they try to retain as much as possible the original between scale covariances present in the ensemble covariances while applying different amounts of spatial localization for different ranges of spatial scales. We start with the normalized ensemble member perturbations, that

are the differences from the ensemble mean divided by $\sqrt{N_{ens}-1}$, where N_{ens} is the total number of ensemble members, and decomposed into a set of overlapping spectral wavebands, with the scale denoted by the index $j = 1, 2, \ldots, J$, as

$$\mathbf{e}_{j,k} = \mathbf{\Psi}_j \mathbf{e}_k, \tag{20.145}$$

where $\mathbf{e}_k$ is the k-th normalized ensemble perturbation, $\mathbf{\Psi}_j$ is the spectral filter that isolates the j-th spectral waveband, and $\mathbf{e}_{j,k}$ is the k-th ensemble perturbation containing only the j-th waveband.

To retain the between-scale covariances, the filtered ensemble perturbations are not treated as independent, and so the contribution from each of J scales for each ensemble perturbation is concatenated to create extended vectors, and the resulting spatial/spectral ensemble covariance matrix can be written as

$$\mathbf{B}_{SS} = \sum_{k=1}^{N_{ens}} \begin{bmatrix} \mathbf{e}_{1,k} \\ \mathbf{e}_{2,k} \\ \vdots \\ \mathbf{e}_{J,k} \end{bmatrix} \begin{bmatrix} \mathbf{e}_{1,k}^T & \mathbf{e}_{2,k}^T & \cdots & \mathbf{e}_{J,k}^T \end{bmatrix}. \tag{20.146}$$

It is stated in [50] that the original ensemble covariance matrix can be obtained by simply summing the contributions from each scale, including the between-scale covariances, giving

$$\mathbf{B}_{ens} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots \mathbf{I} \end{bmatrix} \mathbf{B}_{SS} \begin{bmatrix} \mathbf{I} \\ \mathbf{I} \\ \vdots \\ \mathbf{I} \end{bmatrix} = \sum_{k=1}^{N_{ens}} \left(\sum_{j1=1}^{J} \mathbf{e}_{j1,k} \right) \left(\sum_{j2=1}^{J} \mathbf{e}_{j2,k}^T \right) = \sum_{k=1}^{N_{ens}} \mathbf{e}_k \mathbf{e}_k^T, \tag{20.147}$$

where $j1$ and $j2$ denote the indices for each possible pair of scales. We should note here that this a condition for this to work and that is that for each wavenumber the filter functions must sum to one.

From [50] the spatial/spectral covariance matrix can be rewritten in terms of submatrices representing the spatial covariances for each possible pair of scales $(j1, j2)$, as in

$$\mathbf{B}_{SS,j1,j2} = \sum_{k=1}^{N_{ens}} \mathbf{e}_{j1,k} \mathbf{e}_{j2,k}^T. \tag{20.148}$$

As such scale-dependent spatial covariance localization can be applied to each submatrix by performing a Schur product with a spatial localization matrix, $\mathbf{L}$, that varies as a function of the waveband indices $j1$ and $j2$:

$$\mathbf{B}_{SS_{loc},j1,j2} = \mathbf{B}_{SS,j1,j2} \circ \mathbf{L}_{j1,j2}. \tag{20.149}$$

The covariance matrix with scale-dependent spatial localization is transformed back into the spatial domain by summing all of the localized submatrices:

$$\mathbf{B}_{sd_loc} = \sum_{j1=1}^{J} \sum_{j2=1}^{J} \mathbf{B}_{ss,j1,j2} \circ \mathbf{L}_{j1,j2}. \tag{20.150}$$

The next part of [50] is associated with applying this technique to EnVAR, where to ensure that the complete spatial/spectral localization matrix is positive semi-definite, and to facilitate implementing this in to an EnVAR system that uses a square root of **B** and its transpose as a preconditioner, the spatial localization matrix for a given pair of scales $(j1, j2)$ is defined as

$$\mathbf{L}_{j1,j2} = \left(\mathbf{L}_{j1,j1}\right)^{\frac{1}{2}} \left(\mathbf{L}_{j2,j2}\right)^{\frac{T}{2}}. \tag{20.151}$$

As a result of this the complete spatial/spectral localization matrix can be written as

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{1,1} & \mathbf{L}_{1,2} & \cdots & \mathbf{L}_{1,J} \\ \mathbf{L}_{2,1} & \mathbf{L}_{2,2} & \cdots & \mathbf{L}_{2,J} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{J,1} & \mathbf{L}_{J,2} & \cdots & \mathbf{L}_{J,J} \end{pmatrix} = \begin{pmatrix} \mathbf{L}_{1,1}^{\frac{1}{2}} \\ \mathbf{L}_{2,2}^{\frac{1}{2}} \\ \vdots \\ \mathbf{L}_{J,J}^{\frac{1}{2}} \end{pmatrix} \begin{pmatrix} \mathbf{L}_{1,1}^{\frac{T}{2}} & \mathbf{L}_{2,2}^{\frac{T}{2}} & \cdots & \mathbf{L}_{J,J}^{\frac{T}{2}} \end{pmatrix}. \tag{20.152}$$

This enables the following control variable transform implemented in an existing EnVAR system for computing the analysis increment, $\Delta\mathbf{x}$, from the control vector, denoted by $\boldsymbol{\xi}_k$, for the portion corresponding to the k-th ensemble perturbation:

$$\Delta\mathbf{x} = \begin{bmatrix} \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \end{bmatrix} \sum_{k=1}^{N_{ens}} \begin{bmatrix} \mathbf{e}_{1,k} \\ \mathbf{e}_{2,k} \\ \vdots \\ \mathbf{e}_{J,k} \end{bmatrix} \circ \left(\begin{bmatrix} \mathbf{L}_{1,1}^{\frac{1}{2}} \\ \mathbf{L}_{2,2}^{\frac{1}{2}} \\ \vdots \\ \mathbf{L}_{J,J}^{\frac{1}{2}} \end{bmatrix} \boldsymbol{\xi}_k \right) \equiv \sum_{k=2}^{N_{ens}} \mathbf{e}_k \circ \left(\mathbf{L}^{\frac{1}{2}} \boldsymbol{\xi}_k \right) \tag{20.153}$$

In [48] they present results using this technique with a one-dimensional domain, as well as with a sea-ice EnVAR data assimilation system with promising results. What we show here is from [56] where they have implemented this approach into the global deterministic weather prediction system.

In this implementation, the authors arbitrarily chose three wave bands due to the computational costs increase from this approach. We have a copy of figure 1 from [56] that shows the spectral filter coefficients used to separate background error covariances in to three horizontal bands in Fig. 20.4.

The filter response function to isolate the scales are supposed to represent these associated with synoptic, subsynoptic and mesoscales, where for the large scales the response function is equal to 1 for wave numbers 0 to 4, wavelength of 10000 km, and then decays following the square of a cosine to 0 at wavenumber 20, equivalent to 2000 km. For the small scales the response function is 0 up to wavenumber 20 and reaches a plateau of 1 starting at wavenumber 80, 500 km. The median scale response function equal to the differences between the value 1 and the sum of the two other functions. This is to ensure the condition that the three overlapping response function sum to 1, as mentioned in the theory presented earlier.

In Fig. 20.5 we have a copy of figure 2 from [56] which is an example of ensemble perturbations for temperature on a model level approximately 700 hPa in the top panel, and then the three lower panels are the scale decomposition perturbations and were obtained by first transforming the original ensemble member into spectral space, followed by multiplying the resulting spectral coefficients by the

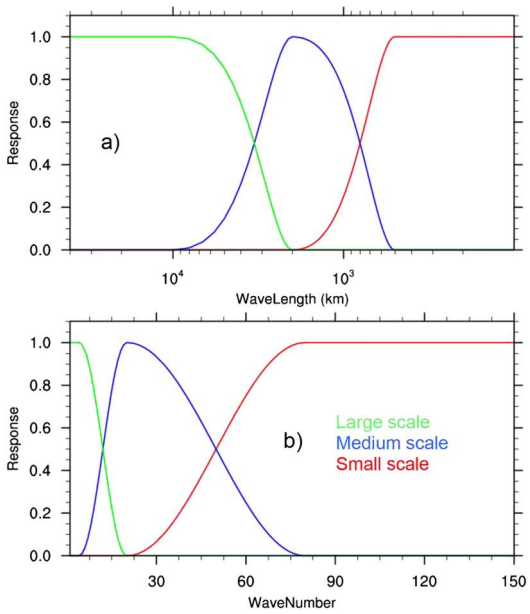


FIGURE 20.4

Copy of figure 1 from Caron, J., and Buehner, M. (2018). Scale-Dependent Background Error Covariance Localization: Evaluation in a Global Deterministic Weather Forecasting System, Monthly Weather Review, 146(5), 1367-1381. **© American Meteorological Society. Used with permission.** [56].

filter describe above, and then transforming back into grid point space. The scales of these perturbation indicates that this approach is a reasonable first attempt. The reader is referred to [56] for more details and results with this approach.

20.9 Ensemble Kalman Smoother

As with the Kalman smoother, there are many different versions of the ensemble Kalman smoother. However, we shall briefly present the theory for an ensemble square root smoother (EnSRS), which is based on the theory of the ensemble square root Kalman filter and the fixed-lag Kalman smoother from Cohn et al. [70], and is from an American Meteorological Society's annual meetings proceedings by Whitaker and Compo [471].

The starting point is to understand that the fixed-lag Kalman smoother is a means of providing retrospective analysis capability into data assimilation. It introduces the lag l, such that for $l = 0$, we obtain the Kalman filter equations. Whitaker and Compo introduce the ensemble square root version of the fixed-lag Kalman smoother from [70]. In the terminology of the ensemble approach, then the overbar represents the ensemble average; we also have that the covariances in the following expressions are those of the ensemble approximations, and not the full rank approximations. Therefore, the general

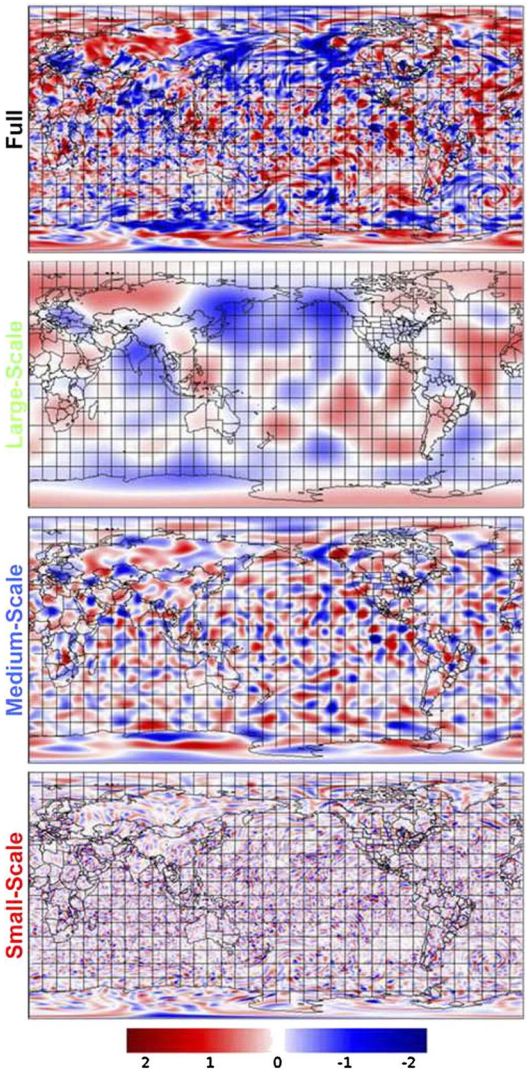


FIGURE 20.5

Copy of figure 2 from Caron, J., and Buehner, M. (2018). Scale-Dependent Background Error Covariance Localization: Evaluation in a Global Deterministic Weather Forecasting System, Monthly Weather Review, 146(5), 1367-1381. **©American Meteorological Society. Used with permission**.

equation for the ensemble fixed-lag Kalman smoother is

$$\bar{\mathbf{x}}^{a}_{k|\,k+l} = \bar{\mathbf{x}}^{a}_{k|\,k+l-1} + \boldsymbol{K}_{k\,|k+l}\left(\bar{\mathbf{y}}_{k+l} - \mathbf{H}_{\mathbf{k+l}}\bar{\mathbf{x}}^{\mathbf{b}}_{\mathbf{k+l}|\,\mathbf{k+l-1}}\right), \tag{20.154}$$

where $\boldsymbol{K}_{k\,|k+l}$ is referred to as the EnSRS's gain matrix and is defined as

$$\boldsymbol{K}_{k\,|k+l} = \left(\mathbf{H_k P^{fa}_{k+l,k|k+l-1}}\right)^T \left(\mathbf{H_{k+l} P^{f}_{k+l|k+l-1} H^{T}_{k+l}} + \mathbf{R_{k+l}}\right)^{-1}. \tag{20.155}$$

The matrix $\mathbf{P^{fa}_{k+l,k|k+l-1}}$ is referred to as the forecast-analysis error cross covariance matrix between the background field used in the Kalman falter updates equation for time $k+l$ and the lag $l-1$ Kalman smoother analysis at time k. It is stated in Whitaker and Compo that it is possible to approximate this matrix through the ensemble, as

$$\left(\mathbf{H_k P^{fa}_{k+l,k|k+l-1}}\right)^T = \overline{\mathbf{x}^{a'}_{k\,|k+l-1}\left(\mathbf{H_{k+l}} \mathbf{x}^{\mathbf{b}'}_{\mathbf{k+l|k+l-1}}\right)}. \tag{20.156}$$

The lag l ensemble mean analysis is calculated by using (20.154)–(20.156). The lag l analysis for the deviations from the ensemble mean is obtained through

$$\mathbf{x}^{a'}_{k|\,k+l} = \mathbf{x}^{a'}_{k|\,k+l-1} - \widetilde{\boldsymbol{K}}_{k\,|k+l} \mathbf{H_{k+l}} \mathbf{x}^{\mathbf{b}'}_{\mathbf{k+l|\,k+l-1}}, \tag{20.157}$$

where $\widetilde{\boldsymbol{K}}_{k|k+l}$ is defined according to Whitaker and Compo such that the ensemble analysis error cross-covariance satisfies the condition from [70] of

$$\mathbf{P^{a}_{k|k+l}} = \mathbf{P^{a}_{k\,|k+l-1}} - \boldsymbol{K}_{\mathbf{k}\,|\mathbf{k}+\mathbf{l}} \mathbf{H_{k+l}} \mathbf{P^{fa}_{k+l,k|k+l-1}}. \tag{20.158}$$

A more recent form of an ensemble fixed-lag smoother is referred to as the **iterative ensemble Kalman smoother**, and a detailed algorithmic description of this smoother can be found in [42]. The original ensemble Kalman smoother can be found in [113].

20.10 Ensemble Sensitivity

We now consider some useful applications of the ensembles to determine sensitivities. We saw in Chapter 13 that we can use the adjoints of the tangent linear model to determine sensitivities in the model to certain initial conditions. Ancell and Hakim in their seminal paper [6] showed that it is possible to build up these same sensitivities through the use of ensembles. We also saw that adjoints enabled us to obtain observational sensitivities. This leads to what is referred to as **targeted observations**. Again it is shown in [6] that it is possible to obtain target observation guidance through an ensemble.

We start by recalling the expression for the adjoint sensitivity to changes in the initial conditions affect the forecast as measured by the scalar matrix J as

$$\partial J = \left(\frac{\delta J}{\partial \mathbf{x}_0}\right)^T \delta \mathbf{x}_0, \tag{20.159}$$

where the term in the brackets is the adjoint sensitivity with respect to the initial state.

To derive the ensemble sensitivity, we right multiply (20.159) by $\delta \mathbf{x}_0^T$, and apply the expectation operator, which results in

$$\mathbb{E}\left[\delta J \delta \mathbf{x}_0^T = \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T \delta \mathbf{x}_0 \delta \mathbf{x}_0^T\right]. \tag{20.160}$$

We should note that the gradient term in (20.160) is a deterministic quantity applying to a control trajectory, and as such we can take it out of the expectation operator, which leaves

$$\mathbb{E}\left[\delta J \delta \mathbf{x}_0^T\right] = \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T \mathbf{A}, \tag{20.161}$$

where $\mathbb{E}\left[\delta J \delta \mathbf{x}_0^T\right]$ is the covariance of the forecast metric with the initial conditions and $\mathbf{A} = \mathbb{E}\left[\delta \mathbf{x}_0 \delta \mathbf{x}_0^T\right]$ is the **initial-time error covariance matrix**.

Now there are two possible approaches to build a linear regression relationship between the response function and the initial states. The first is to take the inverse of $\mathbf{A}$ such that we would have a multivariate linear regression of the form

$$\mathbb{E}\left[\delta J \delta \mathbf{x}_0^T\right] \mathbf{A}^{-1} = \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T. \tag{20.162}$$

Quite often $\mathbf{A}$ will be too large or too ill-conditioned to invert, and as such [6] suggests using a univariate linear regression of the form

$$\left(\frac{\partial J_e}{\partial \mathbf{x}_0}\right)^T = \mathbb{E}\left[\delta J \delta \mathbf{x}_0\right] \mathbf{D}^{-1}, \tag{20.163}$$

where $\mathbf{D}$ is a diagonal matrix with initial time error variances, and $\left(\frac{\partial J_e}{\partial \mathbf{x}_0}\right)$ is the ensemble sensitivity. Using (20.161), we can show that the relationship for this definition of the ensemble sensitivity can be related to the adjoint sensitivity as

$$\frac{\partial J_e}{\partial \mathbf{x}_0} = \mathbf{D}^{-1} \mathbf{A} \frac{\partial J}{\partial \mathbf{x}_0}. \tag{20.164}$$

As with the other ensemble-based methods, we are approximating an integral of the expectation with an ensemble of size K, and we have the mean corrected ensemble perturbation matrix $\delta \mathbf{X}$; this then implies that

$$\mathbf{A} = \frac{1}{K-1} \delta \mathbf{X} \delta \mathbf{X}^{\mathbf{T}}. \tag{20.165}$$

Ancell and Hakim next turn the ensemble sensitivity measure to the important problem of targeted observations. In their approach they are using the Kalman filter as the analysis update scheme, and for the ensemble of size K, the ensemble estimate response function is represented as a row vector $\mathbf{J}$. If we remove the mean from $\mathbf{J}$, then this provides a set of ensemble perturbations $\delta \mathbf{J}$, that has variance

$$\sigma^2 = \frac{1}{K-1} \delta \mathbf{J} \delta \mathbf{J}^{\mathbf{T}}. \tag{20.166}$$

If we now substitute the expression in (20.159) for $\delta \mathbf{J}$ and replace $\delta \mathbf{x}_0$ by $\delta \mathbf{X}_\mathbf{0}$, then we obtain the definition for the ensemble estimate of $\mathbf{A}$, and the variance of the response function becomes

$$\sigma^2 = \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T \mathbf{A} \left(\frac{\partial J}{\partial \mathbf{x}_0}\right). \tag{20.167}$$

Now when all of the observations are assimilated, the background error covariance matrix, $\mathbf{B}$, is updated through $\mathbf{A} = (\mathbf{I} - \boldsymbol{K}\mathbf{H})\,\mathbf{B}$, where $\boldsymbol{K}$ is the usual Kalman gain matrix.

We now let $\mathbf{A}$ be the analysis error covariance matrix after the assimilation of a routine observation, and let $\mathbf{A}'$ represent the analysis error covariance matrix after the assimilation of the targeted observation. The reduction in the response function variance resulting from the assimilation of the new observation is

$$\begin{aligned}\delta\sigma^2 &= \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T (\mathbf{A} - \mathbf{A}') \left(\frac{\partial J}{\partial \mathbf{x}_0}\right), \\ &= \left(\frac{\partial J}{\partial \mathbf{x}_0}\right)^T \mathbf{A}\mathbf{H}^{\mathbf{T}}\mathbf{E}^{-\mathbf{1}}\mathbf{H}\mathbf{A} \left(\frac{\partial \mathbf{J}}{\partial \mathbf{x_0}}\right),\end{aligned}$$

where $\mathbf{E}$ is the innovation error covariance matrix $\mathbf{E} = \left(\mathbf{H}\mathbf{A}\mathbf{H}^{\mathbf{T}} + \mathbf{R}\right)$.

This ensemble metric has been used in several studies now, and for the complete results of the targeted observations experiment see [6]. However, since the first edition this approach has been adapted to form the **ensemble forecast sensitivity to observations**, which in turn has been adapted to form a proactive quality control measure, that we present now.

20.11 Ensemble Forecast Sensitivity to Observations (EFSO)

As we just mentioned, EFSO is based upon the research and development of Ancell and Hakim. The starting point is the analysis equation

$$\overline{\mathbf{x}}_0^a - \overline{\mathbf{x}}_0^b = \mathbf{K}\delta\overline{\mathbf{y}}_0^{ob}, \tag{20.168}$$

where $\overline{\mathbf{x}}_0^a$ and $\overline{\mathbf{x}}_0^b$ are the ensemble mean analysis and background respectively, and $\delta\overline{\mathbf{y}}_0^{ob}$ is the observation minus background innovation of the ensemble mean, all valid at time $t = 0$.

In the EnKF it is possible to directly estimate the Kalman gain matrix, $\mathbf{K}$, by

$$\mathbf{K} = \mathbf{A}\mathbf{H}^T\mathbf{R}^{-1} \approx \frac{1}{K-1}\left(\mathbf{X}^a\mathbf{X}^{a^T}\right)\mathbf{H}^T\mathbf{R}^{-1} \approx \frac{1}{K-1}\mathbf{X}^a\mathbf{Y}^{a^T}\mathbf{R}^{-1}, \tag{20.169}$$

where K is the ensemble size, $\mathbf{A}$ is the analysis error covariance matrix, $\mathbf{X}^a$ is the matrix of the analysis perturbations valid at time $t = 0$; $\mathbf{Y}^a = \mathbf{H}\mathbf{X}^a$. Thus the analysis equation can now be written as

$$\overline{\mathbf{x}}_0^a - \overline{\mathbf{x}}_0^b \approx \frac{1}{K-1}\mathbf{X}^a\mathbf{Y}^{a^T}\mathbf{R}^{-1}\delta\overline{\mathbf{y}}_0^{ob}. \tag{20.170}$$

The next step is to follow the technique from [237] where we measure the change of the forecast error due to assimilation by

$$\begin{aligned}(\Delta\mathbf{e})^2 &= \mathbf{e}_{t|0}^T\mathbf{C}\mathbf{e}_{t|0} - \mathbf{e}_{t|-6}^T\mathbf{C}\mathbf{e}_{t|-6}, \\ &= \left(\mathbf{e}_{t|0} - \mathbf{e}_{t|-6}\right)^T\mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right),\end{aligned} \tag{20.171}$$

where

$$\mathbf{e}_{t|0} \equiv \overline{\mathbf{x}}_{t|0} - \mathbf{x}_t^v, \quad \text{and} \quad \mathbf{e}_{t|-6} \equiv \overline{\mathbf{x}}_{t|-6} - \mathbf{x}_t^v, \tag{20.172}$$

where $\overline{\mathbf{x}}_{t|-6}$ and $\overline{\mathbf{x}}_{t|0}$ represent the ensemble mean forecast valid at time t initialized respectively, at time -6 and 0, $\mathbf{x}t^v$ is the verifying state at time t, and $\mathbf{C}$ defines the energy norm, as we saw with the FSOI earlier. Combining all the features presented so far here, with the usual notation for the nonlinear and linearized numerical models, we obtain:

$$\begin{aligned}(\Delta\mathbf{e})^2 &= \left(\overline{\mathbf{x}}_{t|0} - \overline{\mathbf{x}}_{t|-6}\right)^T \mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right), \\ &\approx \left[\mathbf{M}_{t|0}\left(\overline{\mathbf{x}}_0^a - \overline{\mathbf{x}}_0^b\right)\right]^T \mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right), \\ &\approx \frac{1}{K-1}\left(\mathbf{M}_{t|0}\mathbf{X}^a\mathbf{Y}^{aT}\mathbf{R}^{-1}\delta\overline{\mathbf{y}}_0^{ob}\right)^T \mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right), \\ &\approx \delta\overline{\mathbf{y}}_0^{ob^T}\frac{1}{K-1}\mathbf{R}^{-1}\mathbf{Y}^a\mathbf{X}_{t|0}^{f^T}\mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right),\end{aligned} \tag{20.173}$$

where $\mathbf{X}_{t|0}$ is the matrix of forecast perturbations initialized at time 0 and valid at time t. (20.173) can be interpreted as an inner product of the O-B innovation vector, $\delta\overline{\mathbf{y}}_0^{ob}$, and a sensitivity vector

$$(\Delta\mathbf{e})^2 \approx \delta\overline{\mathbf{y}}_0^{ob}\frac{\partial\Delta\mathbf{e}^2}{\partial\mathbf{y}},$$

where

$$\frac{\partial\left(\Delta\mathbf{e}\right)^2}{\partial\mathbf{y}} = \frac{1}{K-1}\mathbf{R}^{-1}\mathbf{Y}^a\mathbf{X}_{t|0}^{f^T}\mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right).$$

It is then stated in [181] that the change in the forecast error due to the assimilation of observation $\mathbf{y}_0^o$ can be decomposed into a sum of contributions from the O-B innovations associated with each observation. Thus for the l-th element of $\mathbf{y}_0^o$, its contribution from the corresponding O-B innovation, $\delta\overline{y}_{0,l}^{ob}$, is

$$(\Delta\mathbf{e})^2\Big|_{y_{0,l}^0} \approx \delta\overline{y}_{0,l}^{ob}\frac{\partial\left(\Delta\mathbf{e}\right)^2}{\partial\mathbf{y}}. \tag{20.174}$$

The expression in (20.174) is the definition of the EFSO.

As with any EnKF, localization needs to be applied to the error covariances to suppress sampling errors whenever K is smaller than the number of degrees of freedom of the predicted dynamical system. After applying localization, the sensitivity vector becomes

$$\frac{\partial\left(\Delta\mathbf{e}\right)^2}{\partial y} = \frac{1}{K-1}\mathbf{R}^{-1}\left[\boldsymbol{\rho}\circ\left(\mathbf{Y}^a\mathbf{X}_{t|0}^{f^T}\right)\right]\mathbf{C}\left(\mathbf{e}_{t|0} + \mathbf{e}_{t|-6}\right), \tag{20.175}$$

where $\boldsymbol{\rho}$ is a matrix whose (i, j) elements is a localization factor of the l-th observation onto the j-th grid point.

Unlike in 4D VAR, in EnKF an explicit estimation of the analysis error covariance and gain matrices are available. This can be used to approximately estimate how the analysis and forecast would change by not assimilating a given subset of the observations. Let $\delta\overline{\mathbf{y}}_0^{ob,deny}$ be a column vector whose elements corresponding to the denied observations are identical to those of $\delta\overline{\mathbf{y}}_0^{ob}$ but others are all set to zero. Then, assuming that the Kalman gain $\mathbf{K}$ does not change much by excluding the denied observations

(which should be valid if the fraction of denied observations is small), the analysis that would be obtained by not assimilating them can be approximated

$$\overline{\mathbf{x}}_0^{a,deny} \approx \overline{\mathbf{x}}_0^b + \mathbf{K}\left(\delta\overline{\mathbf{y}}_0^{ob} - \delta\mathbf{y}_0^{ob,deny}\right) = \overline{\mathbf{x}}_0^a - \mathbf{K}\delta\overline{\mathbf{y}}_0^{ob,deny}. \tag{20.176}$$

Applying localization and using the expression in (20.169), we obtain

$$\overline{\mathbf{x}}_0^{a,deny} - \overline{\mathbf{x}}_0^a \approx -\frac{1}{K-1}\left[\boldsymbol{\rho} \circ \mathbf{X}^a\mathbf{Y}^{a^T}\right]\mathbf{R}^{-1}\delta\overline{\mathbf{y}}_0^{ob,deny}. \tag{20.177}$$

A similar impact on the forecast can also be measured through

$$\overline{\mathbf{x}}_0^{f,deny} - \overline{\mathbf{x}}_0^f \approx -\frac{1}{K-1}\left[\boldsymbol{\rho} \circ \mathbf{X}_{t|0}^f\mathbf{Y}^{a^T}\right]\mathbf{R}^{-1}\delta\overline{\mathbf{y}}_0^{ob,deny}. \tag{20.178}$$

Note that the 0-h EFSO, or the analysis sensitivity to observations, can also be used to retrospectively check whether the inconsistent observations were effectively rejected. Verifying against the analysis using (20.171) yields

$$(\Delta\mathbf{e})^2 = -\left(\mathbf{K}\delta\overline{\mathbf{y}}_0^{ob}\right)^T\mathbf{C}\delta\overline{\mathbf{x}}^{a,b}, \tag{20.179}$$

from which it is possible to deduce that the contribution to $(\Delta\mathbf{e})^2$ from a single observation is negative (positive) if the partial analysis increment attributes to that observation is consistent (inconsistent) with the total analysis increment. Therefore, it is expected that the 0-h EFSO should be mainly negative, provided that the nonlinear quality control has worked.

The rest of [181] presents a description of the steps in the quality control procedure, and the reader is referred to this paper for more of the details.

Another advancement in ensemble data assimilation since the first edition is the introduction of the local ensemble tangent linear model, which we present in the next section.

20.12 Local Ensemble Tangent Linear Model (LETLM)

The derivation that we present here comes from [39], where the motivation for this work is to be an enabler for coupled 4D VAR, we will go into more details about coupled data assimilation in the applications chapter.

The theory starts by stating that in finite difference and finite volume models, the change in a variable's value over a time step is determined by the values of variables in a very near neighborhood of the variable in question. The precise number of variables contributing to the time change depends on the order of the finite difference approximations to spatial derivatives and to the type of time-stepping scheme employed. This is heuristically shown in Figure 3 from [39] which is a grid of temperature points in 2D and shows a circle around the center point to indicate which points have influence over the center one, which defines the LETLM influence volume. This influence volume contains, at a minimum, all of the variables that will influence the future time state of the variable at the center of the influence volume.

Now if we suppose that there are n variables at time t in the influence volume that influence the later time $t + \delta t$ values of the central m-th model variable, denoted $x_m\,(t + \delta t)$, and suppose that at the initial

time t we have a K-member ensemble forecast that has a small mean square distance $\sigma^2_{\max}$ around the time t influence volume guess field $\left(x_1^g, x_2^g, \cdots, x_n^g\right)$. Provided that the influence region contains all the variables used by nonlinear model to update the m-th variable

$$\begin{aligned}&\left[x_{m1}\left(t+\delta t\right), x_{m2}\left(t+\delta t\right), \ldots, x_{mK}\left(t+\delta t\right)\right] = \\ &\left\{\mathcal{M}_m\left(\begin{bmatrix} x_1^g+\delta x_{11} \\ x_2^g+\delta x_{21} \\ \vdots \\ x_N^g+\delta x_{n1}\end{bmatrix}\right), \mathcal{M}_m\left(\begin{bmatrix} x_1^g+\delta x_{1} \\ x_2^g+\delta x_{22} \\ \vdots \\ x_N^g+\delta x_{n2}\end{bmatrix}\right), \cdots, \mathcal{M}_m\left(\begin{bmatrix} x_1^g+\delta x_{1K} \\ x_2^g+\delta x_{2K} \\ \vdots \\ x_N^g+\delta x_{nK}\end{bmatrix}\right)\right\},\end{aligned} \tag{20.180}$$

where $\mathcal{M}_m$ is the part of the nonlinear model that determines the future time state of the m-th variable from the n variables in the influence volumes, and $\left(x_1^g+\delta x_{ij}, x_2^g+\delta x_{2j}, \cdots, x_n^g+\delta x_{nj}\right)$ is the j-th perturbed of the true values at time t within the n-variable influence region.

Tangent linear models, as we were saw earlier, are defined as perturbations to a guess trajectory. In this case the guess trajectory corresponding to each ensemble perturbation is given by

$$\begin{aligned}&\left[x_m^g\left(t+\delta t\right), x_m^g\left(t+\delta t\right), \ldots, x_m^g\left(t+\delta t\right)\right] = \\ &\left\{\mathcal{M}_m\left(\begin{bmatrix} x_1^g \\ x_2^g \\ \vdots \\ x_N^g\end{bmatrix}\right), \mathcal{M}_m\left(\begin{bmatrix} x_1^g \\ x_2^g \\ \vdots \\ x_N^g\end{bmatrix}\right), \cdots, \mathcal{M}_m\left(\begin{bmatrix} x_1^g \\ x_2^g \\ \vdots \\ x_N^g\end{bmatrix}\right)\right\},\end{aligned} \tag{20.181}$$

It is then stated that without loss of generality, the local nonlinear model can be represented in terms of a Taylor expansion of operators on the ensemble perturbations and so in the limit of vanishingly small ensemble perturbations, when $\sigma^2_{\max} \to 0$, only the linear operator within this expansion is non-negligible and subtracting (20.181) from (20.180) yields

$$\left[\delta_{m1}\left(t+\delta t\right), \delta_{m2}\left(t+\delta t\right), \ldots, \delta_{mK}\left(t+\delta t\right)\right] = \mathbf{M}_m\left(i+1, i\right)\begin{bmatrix} \delta x_{11}, \delta x_{12}, \ldots, \delta x_{1K} \\ \delta x_{21}, \delta x_{22}, \ldots, \delta x_{2K} \\ \vdots \\ \delta x_{n1}, \delta x_{n2}, \ldots, \delta x_{nK}\end{bmatrix}, \tag{20.182}$$

where $\delta x_{mj}\left(t+\delta t\right) = x_{mj}\left(t+\delta t\right) - x_m^g\left(t+\delta t\right)$ and where $\mathbf{M}\left(i+1, i\right)$ is the true local tangent linear model for the m-th grid point for the evolution of a perturbation form the i ith time step to the $(i+1)$-th time step, and $\mathbf{M}_m\left(i+1, i\right)$ has one row and n columns.

The next part of the derivation is to define

$$\begin{aligned}\boldsymbol{\Xi}_m &\equiv \left[\delta_{m1}\left(t+\delta t\right), \delta_{m2}\left(t+\delta t\right), \ldots, \delta_{mK}\left(t+\delta t\right)\right], \\ \mathbf{X}_m &\equiv \begin{bmatrix} \delta x_{11}, \delta x_{12}, \ldots, \delta x_{1K} \\ \delta x_{21}, \delta x_{22}, \ldots, \delta x_{2K} \\ \vdots \\ \delta x_{n1}, \delta x_{n2}, \ldots, \delta x_{nK}\end{bmatrix},\end{aligned} \tag{20.183}$$

which implies that (20.182) can be written as

$$\mathbf{\Xi}_m = \mathbf{M}_m\,(i+1,i)\,\mathbf{X}_m. \tag{20.184}$$

Note that if the $n \times n$ matrix $\left(\mathbf{X}_m\mathbf{X}_m^T\right)$ has an inverse then right-multiplying (20.184) by $\mathbf{X}_m^T$ and then solving the resulting equation for $\mathbf{M}_m$ gives

$$\mathbf{M}_m\,(i+1,i) = \mathbf{\Xi}_m\mathbf{X}_m^T\left(\mathbf{X}_m\mathbf{X}_m^T\right)^{-1}. \tag{20.185}$$

The inverse $\left(\mathbf{X}_m\mathbf{X}_m^T\right)^{-1}$ is guaranteed to exist provided that the matrix $\mathbf{X}_m$ has n non-zero singular values. In a well formed ensemble, $K-1$ ensemble members will be linearly independent. Hence, with well-formed ensemble members (20.185) will precisely recover the true local tangent linear model $\mathbf{M}_m\,(i+1,i)$ when $K \geq n+1$. In strong-constraint 4D VAR, the tangent linear model is only used to propagate perturbations that lie in the vector sub-space spanned by the eigenvectors of the background error covariance matrix $\mathbf{B}$ that have non-zero eigenvalues, [39]. If the corresponding error sub-space in the influence region can be spanned with just q vectors where $q < n$, then well-formed ensembles with just $K \geq q+1$ would be sufficient to perfectly describe the true local tangent linear model. Thus, a perfect ensemble tangent linear model may be possible with fewer ensemble members K than influence region variables n when $\mathbf{B}$ has zero eigenvalues. If $K < n+1$ then the inverse in (20.185) would need to be replaced by a pseudo-inverse; furthermore, with $K < n+1$ an equivalent but more computationally efficient form of (20.185) is

$$\mathbf{M}_m\,(i+1,i) = \mathbf{\Xi}\left(\mathbf{X}_m^T\mathbf{X}_m\right)^{-1}\mathbf{X}_m^T. \tag{20.186}$$

Because the ensemble perturbation matrix $\mathbf{X}_m$ only pertains to model variables within the spatially local influence volume associated with x_m, the ensemble-based TLMs in either (20.185) or (20.186) as **Local Ensemble Tangent Linear Models** or LETLMs for short.

In [39] they present a discussion about the limits of the linearization assumption, and provide an alternative in this limit. There is also a discussion about how to implement this approach. We finish this summary with the description of the adjoint of this LETLM.

Therefore, let $\mathbf{S}_m$ be the matrix that maps the global state L-vector $\mathbf{x}$ to the list of variables in the n-vector $\mathbf{x}_m$ defining the influence volume required to evolve the m-th model variable over a single time step so that

$$\mathbf{x}_m = \mathbf{S}_m\mathbf{x}. \tag{20.187}$$

$\mathbf{S}_m$ has as many rows as there are variables in the m-th influence volume and as many columns as there are variables in the global state vector. There is only one element in each row of $\mathbf{S}_m$ that is not equal to zero and this element is equal to unity. If we now let the index j be the index of the column of the i-th row of $\mathbf{S}_m$ that is equal to unity; then the i-th row maps the j-th element of the vector $\mathbf{x}$ to the i-th element of the local vector $\mathbf{x}_m$. Hence, the j-th row of $\mathbf{S}_m^T$ in the operation $\mathbf{S}_m^T\mathbf{x}_m$ maps the i-th element of the local vector $\mathbf{x}_m$ back to the row of the global state vector from which it originally came. Hence, the operation $\mathbf{S}_m^T\mathbf{x}_m$ puts all of the variables in $\mathbf{x}_m$ onto the positions in the global state vector to which they originally belonged. Thus, to apply the adjoint of the LETLM to an arbitrary global L-vector $\mathbf{z}$ with elements $\mathbf{z}^T = [z_1, z_2, \ldots, z_m, \ldots, z_L]$, we use

$$\mathbf{M}\,(i+1,i)^T\,\mathbf{z} = \sum_{m=1}^{L} \mathbf{S}_m^T \mathbf{M}_m\,(i+1,i)^T\,z_m, \tag{20.188}$$

where $\mathbf{M}\,(i+1,i)^T$ denotes the adjoint of the global tangent linear model implied by the complete set of LETLMs, $\{\mathbf{M}_1.\mathbf{M}_2,\ldots,\mathbf{M}_L\}$ that are defined by (20.185), where each LETLM is a local row vector and hence the $\mathbf{M}_m\,(i+1,i)^T$ in (20.188) is a local column vector. We should note here that L-n of the rows in are entirely filled with zeros. Hence, we only has to consider the effect of the n-rows of $\mathbf{S}_m^T$ that are non-zero and hence the operation $\mathbf{S}_m^T\mathbf{M}_m^T$ is rapid as it amounts to simply assigning the correct global index to the n-variables listed in the local influence vector $\mathbf{M}_m^T z_m$.

The rest of [39] presents how it could be used to compute forecast sensitivity as well as present results with a coupled 4D VAR system, and the reader is referred to this paper for these details.

20.13 Summary

In this chapter we have introduced the fast and continually expanding field of ensemble-based data assimilation. We started with the EnKF that first appeared in [111]. We have presented many different versions of the filter: EnSRF, ETKF, LETKF, MLEF, along with the new hybrid variational-ensemble approaches. We have highlighted some of the drawbacks of the ensemble methods: the need for inflation to prevent the filter from diverging, which can also be caused by the undersampling of the smaller scales; having to localized to remove these scales; and having to perturb the observations in a non-square root formulations. We have also referenced an approach to enforce balance through the localization functions.

The advantage of the ensemble-based data assimilation systems is that they allow us to relax the isotropic and homogeneous requirements for the background error covariance, as the estimate that the ensemble members provide is flow dependent. However, we will always have a rank deficient problem and have to ensure that the localization removes the undersampled scales.

We have also shown that it is possible to obtain an ensemble-based equivalent to the adjoint sensitivity, and therefore avoid deriving and coding an adjoint of a possibly highly nonlinear problem. In this edition we have presented how this work has been extended to form an equivalent EFSO of the FSOI, as well as work to introduce the LETLM

As we mentioned earlier, there is a lot of movement, especially in the operational NWP world, to try to remove the adjoint and TLM from the minimization of the 4D VAR cost function. This is not true at ECMWF and Mètèo-France, where they run multiple 4D VARs to create their flow dependency information. One of the advantages of 4D VAR is that it enables us to introduce constraints on to the solution due to the variational formulation, but another point that we need to make here is that it is very difficult for ensemble-based methods to go non-Gaussian. There have been some recent gains in this direction, and we show some of them in the non-Gaussian chapter, where the author has been able to derive a version of the Kalman filter equations, that are the basis of the EKFs, to allow for a full lognormal formulation, as well as a mixed Gaussian-lognormal version, but that these two versions are nonlinear.

As shown in Fletcher and Zupanski [135], the mean of a lognormal distribution is unbounded with respect to growing variance, and as such if we build a minimum variance based approach it would be growing away from the most likely state. However, it is possible for the MLEF to go non-Gaussian, and

since the first edition the update to the covariance matrix for the lognormal distribution has been derived due to the aforementioned new version of the Kalman filter equations for the lognormal distribution.

Why did we bring up the non-Gaussian issue? We add this to introduce our next two chapters, that are associated with data assimilation methods that can cope with non-Gaussian distributions. The first of these starts with the work of the author with lognormal distributions and the mixed distributions in a variational formulation, but includes the progress that has been made in other data assimilation schemes to address non-Gaussianity, including introducing the GIGG filter reference earlier, as well as Gaussian anamorphosis. The second chapter introduces the theory of **Markov chain Monte Carlo and particle filters**, and shows the progress that these approaches have made since the first edition, as well as introducing the local particle filter, particle smoothers, and sigma-point filters.

This page intentionally left blank

CHAPTER

Non-Gaussian Based Data Assimilation

21

Contents

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00025-5

Up to this point, all the theory presented for the derivation of the variational, Kalman Filter, and ensemble-based data assimilation schemes has assumed that the background, observational, and model errors are Gaussian distributed. However, as we shown earlier, the Gaussian distribution is for random variables that can take values between $(-\infty, \infty)$, but if we are dealing with positive definite variables, i.e., $x > 0$, or positive semi-definite, $x \geq 0$, then a Gaussian distribution cannot describe their behavior. Examples of positive definite variables for the atmosphere are humidity, surface pressure, and total precipitable water; for oceanography salinity as well as biochemical components.

The reason to go to a non-Gaussian-fits-all formulation is that if a random variable is close to zero, or is skewed toward a smaller values, then the Gaussian distribution, with its symmetry property, could assign probability to values that are less than or equal to zero, which, say for humidity, would be unphysical. It is not unheard of for data assimilation systems based upon Gaussian distributions to assign unphysical values to a positive definite variables. In it shown in [409] that the author's initial Gaussian-based data assimilation system for ocean biochemical assimilation could assign unphysical values as its analysis. When dealing with an operational numerical weather, or ocean, prediction system, having an unphysical value for the analysis state causes problems for the initialization of the numerical forecasting model. These occurrences are called *dropouts* and can be quite expensive to compensate for, due to the limited time available to perform the data assimilation in operations.

There has been development of non-Gaussian variational-based data assimilation systems involving the lognormal distribution. We introduced the univariate lognormal distribution in Chapter 3, and the multivariate lognormal distribution in Chapter 4. This chapter will present the theory of the lognormal distribution-based full field and incremental 3D and 4D VAR schemes. We shall also present the derivations for the equivalent variational schemes with the mixed Gaussian-lognormal distribution, introduced in Chapter 4.

Since the first edition, there has been advancements in non-Gaussian based data assimilation, and so the title of this chapter is no longer variational based. We shall present new derivations of non-Gaussian versions of the Kalman filter. We shall also introduce Gaussian anamorphosis, as well as the Gamma-Inverse-Gamma-Gaussian (GIGG) filter. However, before we start to derive a lognormal-based variational data assimilation scheme, we require the definition for errors that follow a geometric type distribution.

21.1 Error Definitions

We start by recalling the geometric property of the lognormal distribution from Chapter 4. We stated in that chapter that the product and the ratio of two independent lognormally distributed random variables are lognormal random variables as well. Therefore, we require a geometric-based definition for the errors if we are going to consider a lognormal distribution as an alternative to the Gaussian distribution for variational data assimilation.

The first appearance of the definition for lognormally distributed observational errors appears in [68]. Because the lognormal distribution is not an additive distribution, we cannot use the property that the difference between, or the sum of, two independent lognormally distributed random variables is also a lognormally distributed random variable. In fact there is **no known distribution for the difference between two independent lognormally distributed random variables**; however, it is known that it is neither a Gaussian nor a lognormal distribution. We have plotted the histogram of the difference

between two lognormal samples in Fig. 21.1 to illustrate that the distribution for this random variable is neither Gaussian nor lognormal.

However, as the lognormal distribution is a geometric type, we can use the property that the ratio, and the product, of two independent lognormally distributed random variables is also a lognormally distributed random variable. As mentioned above, the first presentation of lognormally distributed errors in a data assimilation context was presented in [68] for observational errors associated with direct observations of the field. The definition for lognormally distributed observations errors involving the

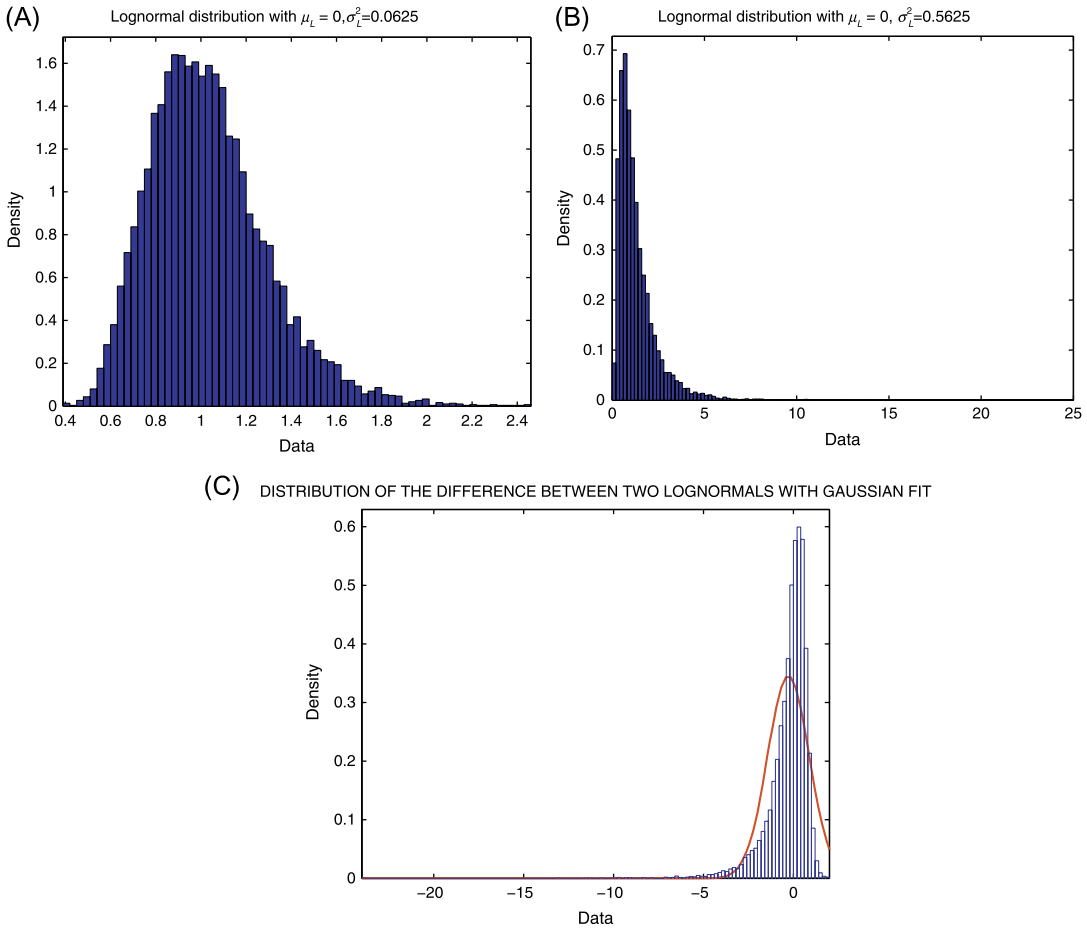


FIGURE 21.1

Plots of two random samples from two different lognormal distributions, and the distribution of the difference between them to illustrate that the distribution of the difference between two lognormal random variables is not a Gaussian.

observation operator appeared in [135] and is given by

$$\boldsymbol{\varepsilon}_i^o \equiv \frac{\mathbf{y_i}}{\mathbf{h}(\mathbf{x})_i}, \quad i = 1, 2, \ldots, N_0. \tag{21.1}$$

The definition for lognormally distributed background errors did not appear until [137], where the geometric property of the lognormal distribution was again utilized. Therefore, the definition for lognormally distributed background errors is given by

$$\boldsymbol{\varepsilon}_{0,j}^b \equiv \frac{\mathbf{x}_{0,j}^t}{\mathbf{x}_{0,j}^b}, \quad j = 1, 2, \ldots, N. \tag{21.2}$$

The definition for lognormally distributed model errors, assuming a one-step Markov process, was defined in [129], where the ratio property was again utilized and results in

$$\boldsymbol{\varepsilon}_n^m \equiv \frac{\mathbf{x}_n^t}{\mathcal{M}_{n-1,n}\left(\mathbf{x}_{n-1}\right)}, \tag{21.3}$$

where $\mathcal{M}_{n,n-1}$ is the nonlinear numerical integration from time t_{n-1} to time t_n and the subscript n represents the time index.

21.2 Full Field Lognormal 3D VAR

In Chapter 16 we presented two types of variational-based data assimilation methods: the full field and incremental versions of both 3D and 4D VAR. The development of the lognormal-based variational systems follows a similar path, where the first lognormal-based variational system was in three dimensions and appears in [135] and was for the full field formulation.

21.2.1 Lognormal Observational Error

The development of the full field 3D VAR for lognormally distributed observational errors, combined with a Gaussian distribution for the background errors, was derived in [135], and is based upon the Bayes's theorem derivation from [259], but now a multivariate lognormal distribution is substituted for the conditional PDF, given the new definition for the observational error in (21.1). The conditional multivariate lognormal distribution using (21.1) is

$$P(\mathbf{y}|\mathbf{x}) \equiv \prod_{i=1}^{N_0} \left(\frac{[\mathbf{h}(\mathbf{x})]_i}{\mathbf{y}_i} \right) \exp\left\{ -\frac{1}{2} \left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right)^T \mathbf{R}_L^{-1} \left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right) \right\}, \tag{21.4}$$

where $\mathbf{R}_L \equiv \mathbb{E}\left[\left(\ln \boldsymbol{\varepsilon}^o - \mathbb{E}\left[\ln \boldsymbol{\varepsilon}^o\right]\right)^2 \right]$.

In [135] there is a discussion about which of the three descriptive statistics; mean, median, or mode, would be best to develop the lognormal-based assimilation scheme around. It was decided in [135] that the mode was the best statistic to use as it is the only one of the three that is bounded with respect to the variance, σ_l^2. The median was rejected as it is non-unique for the multivariate lognormal distribution, while the mean was rejected as it is unbounded with respect to the variance, as well as having to be

solved for componentwise. Another advantage of the mode over the other two descriptive statistics is that it is a function of the covariance matrix. These properties has been presented in Chapter 4 but as a reminder, the mode of a multivariate lognormal distribution is $mode = \exp\{\boldsymbol{\mu} - \langle \boldsymbol{\Sigma}, \mathbf{1} \rangle\}$.

In [135] the lognormal observational condition PDF in (21.4) was combined with the Gaussian-based background error PDF and then the log-likelihood approach was performed, which results in the following cost function to minimize:

$$J(\mathbf{x}) = \frac{1}{2}\left(\mathbf{x}^t - \mathbf{x}^b\right)^T \mathbf{B}^{-1}\left(\mathbf{x}^t - \mathbf{x}^b\right) + \frac{1}{2}(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}))^T \mathbf{R}_L^{-1}(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})) + \left\langle (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})), \mathbf{1}_{N_o} \right\rangle. \tag{21.5}$$

The Jacobian of (21.5) can easily be shown to be

$$\nabla_{\mathbf{x}}(\mathbf{x}) = \mathbf{B}^{-1}\left(\mathbf{x}^t - \mathbf{x}^b\right) - \mathbf{H}^T \mathbf{W}_o^T \left(\mathbf{R}_L^{-1}(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})) + \mathbf{1}_{N_o}\right), \tag{21.6}$$

where

$$\mathbf{W}_o^{-T} \equiv \frac{\partial \ln \mathbf{h}\left(\mathbf{x}^t\right)}{\partial \mathbf{x}} \equiv \begin{pmatrix} \frac{1}{h_1(\mathbf{x})} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & \frac{1}{h_2(\mathbf{x})} & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \frac{1}{h_3(\mathbf{x})} & 0 & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & \frac{1}{h_{N_0}(\mathbf{x})} \end{pmatrix}. \tag{21.7}$$

Looking at the structure of the Jacobian in (21.6), it is quite clear that the derivative of the observational component of the cost function appears similar to a lognormal mode, and that setting (21.6) equal to zero and rearranging results in a nonlinear expression for the true state as

$$\mathbf{x}^t = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T \mathbf{W}_o^{-T}\left(\mathbf{R}_L^{-1}\left(\ln \mathbf{y} - \ln \mathbf{h}\left(\mathbf{x}^t\right)\right) + \mathbf{1}\right). \tag{21.8}$$

We should note that $\mathbf{W}_o^{-T}$ and the observation operator in (21.8) are both functions of $\mathbf{x}^t$. However, it is clear from (21.8) that the state that minimizes (21.5) is a combination of the modes from the background error's Gaussian distribution and the observational error's lognormal distribution.

We showed in Chapter 17.5 that some minimization algorithms require the Hessian of the cost function being minimized, if the dimensionality of the problem enables the evaluation of the inverse of the Hessian matrix. For the cost function in (21.5) the matrix component of the Hessian can easily be shown to be

$$HESS(J(\mathbf{x})) \equiv \mathbf{B}^{-1} + \mathbf{H}^T \mathbf{W}_o^{-T} \mathbf{R}_L^{-1} \mathbf{W}_o^{-1} \mathbf{H}. \tag{21.9}$$

An interesting feature of the Hessian in (21.9) is that is appears to be quite similar to the Gaussian equivalent except for the scalings by the two $\mathbf{W}_o^{-1}$ matrices.

21.2.2 Lognormal Background Errors

The summary here is for the full field variational scheme with lognormally distributed background and observational errors which comes from [137]. This is the situation where we assume that the prior distribution is a lognormal distribution. Therefore, substituting the background error definition from (21.2) into a multivariate lognormal distribution results in the following expression for the prior distribution:

$$P\left(\mathbf{x}^b = \mathbf{x}^t\right) \equiv \prod_{i=1}^{N} \left(\frac{\mathbf{x}^b}{\mathbf{x}^t}\right) \exp\left\{-\frac{1}{2}\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right)^T \mathbf{B}_L^{-1}\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right)\right\}, \tag{21.10}$$

where $\mathbf{B}_L \equiv \mathbb{E}\left[\left(\ln \boldsymbol{\varepsilon}^b - \mathbb{E}\left[\ln \boldsymbol{\varepsilon}^b\right]\right)^2\right]$. To form the posterior distribution we need to multiply (21.10) with (21.4). However, to obtain a cost function for the mode of the posterior distribution we have to take the negative logarithm of this product. Therefore, the cost function for this posterior distribution is

$$\begin{aligned} J(\mathbf{x}) = &\frac{1}{2}\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right)^T \mathbf{B}_L^{-1}\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right) + \left\langle\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right), \mathbf{1}_N\right\rangle \\ &+ \frac{1}{2}\left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right)^T \mathbf{R}_L^{-1}\left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right) + \left\langle\left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right), \mathbf{1}_{N_o}\right\rangle. \end{aligned} \tag{21.11}$$

The Jacobian of (21.11), which comes from [137], can easily be shown to be

$$\nabla_{\mathbf{x}} J(\mathbf{x}) = \mathbf{W}_b^{-T}\left(\mathbf{B}_L^{-1}\left(\ln \mathbf{x}^t - \ln \mathbf{x}^b\right) + \mathbf{1}\right) - \mathbf{H}^T \mathbf{W}_o^T\left(\mathbf{R}_L^{-1}\left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})\right) + \mathbf{1}_{N_o}\right), \tag{21.12}$$

where

$$\mathbf{W}_b^T \equiv \frac{\partial \ln \mathbf{x}^t}{\partial \mathbf{x}^t} \equiv \begin{pmatrix} \frac{1}{x_1^t} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & \frac{1}{x_2^t} & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \frac{1}{x_3^t} & 0 & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & \ddots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & \cdots & \cdots & \frac{1}{x_N^t} \end{pmatrix}. \tag{21.13}$$

To find the associated nonlinear expression for the mode of the posterior distribution, we set (21.12) equal to zero and rearrange to isolate the $\mathbf{x}_t$ term from the background component, which results in

$$\ln \mathbf{x}^t = \underbrace{\ln \mathbf{x}^b - \langle \mathbf{B}_L, \mathbf{1}\rangle}_{\textit{back LN mode}} + \mathbf{B}\mathbf{W}_b^T \underbrace{\mathbf{H}^T \mathbf{W}_o^{-T}\left(\mathbf{R}^{-1}\left(\ln \mathbf{y} - \ln \mathbf{h}\left(\mathbf{x}^t\right)\right) + \mathbf{1}\right)}_{\textit{obs LN mode}}. \tag{21.14}$$

We have indicated the background lognormal mode and the observational lognormal mode in (21.14) to show that the solution appears to be a balance between two lognormal modes. To obtain the non-logarithmic expression for $\mathbf{x}_t$, we simply take the componentwise exponential of (21.14). Thus

$$\mathbf{x}^t = \mathbf{x}^b \circ \exp\left\{-\langle \mathbf{B}_L, \mathbf{1}\rangle + \mathbf{H}^T \mathbf{W}_o^{-T}\left(\mathbf{R}^{-1}\left(\ln \mathbf{y} - \ln \mathbf{h}\left(\mathbf{x}^t\right)\right) + \mathbf{1}\right)\right\}, \tag{21.15}$$

where the operator $\circ$ represents the Hadamard product which is a componentwise multiplication.

As with the lognormally distributed observation errors cost function, we now consider the Hessian of (21.11), which can easily be shown to be

$$HESS\left(J\left(\mathbf{x}^t\right)\right) \equiv \mathbf{W}_b^{-T}\mathbf{B}_L^{-1}\mathbf{W}_b^{-1} + \mathbf{H}^T\mathbf{W}_o^{-T}\mathbf{R}_L^{-1}\mathbf{W}_o^{-1}\mathbf{H}. \tag{21.16}$$

Again, the Hessian matrix in (21.16) looks similar in structure to the Gaussian Hessian matrix, except for the scalings now of the background error and the observational error covariance matrices by the $\mathbf{W}_b^{-1}$ and $\mathbf{W}_o^{-1}$, respectively.

21.3 Logarithmic Transforms

Before the development of the lognormal-based 3D VAR in [135,137], variables that were positive definite were sometimes changed using a logarithmic transform into a new random variable that could take negative values. The advantage of this was that it stopped a positive definite variable from obtaining a negative, unphysical, value from the solution to a Gaussian-based variational data assimilation scheme. Note that the inverse of the logarithmic transform is the exponential, so any negative value for $\log \mathbf{x}$ would invert to $e^{\log \mathbf{x}}$, which would be between 0 and 1 for these values.

In fact, what the logarithmic transform approach does is it implicitly utilizes the property that the natural log of a lognormally distributed random variable is a Gaussian random variable, and therefore can be optimized in a Gaussian data assimilation framework. However, it is shown in [137] that we should be cautious about using this transform, as it does not find the most likely state in lognormal space.

It is shown in [137] that the most likely state found in the Gaussian space inverts back to an estimate of a multivariate median in lognormal space. A worrying feature of the analysis state in Gaussian space inverting to the median in lognormal space is the fact that the lognormal median is independent of variance. **Note**. The difference between the median and the mode increases exponentially as the variance increases, but a second worrying feature of solving a cost function with the logarithmic transform is that the mode in Gaussian space is non-unique when inverted back to lognormal space. In Fig. 21.2 we have a simple illustration of the problems of using this logarithmic transform.

In Fig. 21.2A we have plotted four different univariate lognormal distributions; each one has a different value for the variance, σ^2, but all of them have the same mean μ. In Fig. 21.2B we have plotted the Gaussian distribution that each of the four lognormal distributions transform to through using the logarithmic transform. We can clearly see that all four of these distributions have the same mode in Gaussian space, but we can see from Fig. 21.2A that the four lognormal distributions do not have the same mode, but they do all have the same median. Therefore, we have to be cautious about using the logarithmic transform approach. We say cautious as recent work by Fletcher, Kliewer, and Jones at CIRA, Colorado State University has shown that not all may be lost using the logarithmic approach. We go into more detail about this finding in Section 21.14.

In Table 21.1 we have calculated the difference between the four analysis states from inverting the mode from the transformed space back into lognormal space. As we can see, if the variance is quite small then the median and the mode are almost identical. It is quite well known that for small variances, a lognormal distribution can be approximated by a Gaussian distribution. However, as we can see from the values in Table 21.1, as the variance increases the median begins to drastically overestimate the

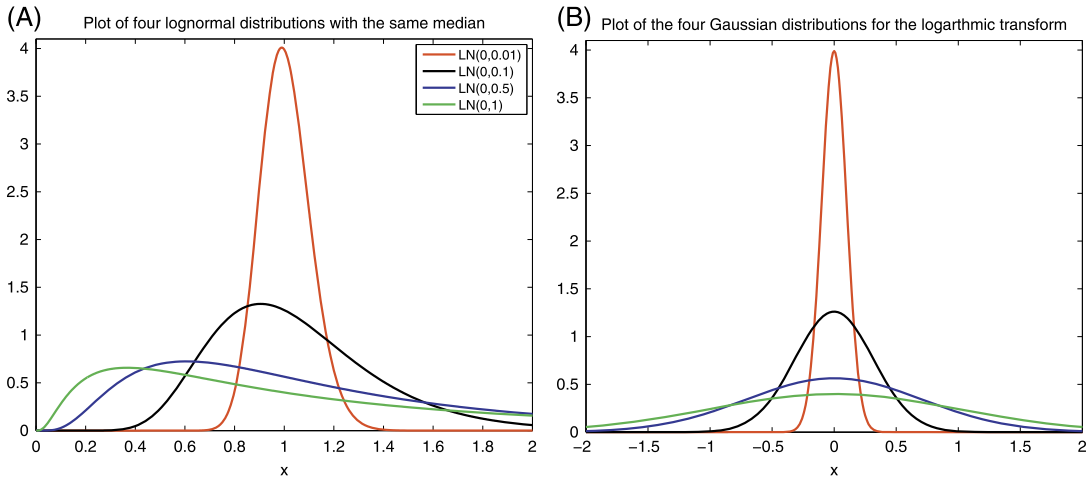


FIGURE 21.2

Plots to illustrate the that higher order moment information is lost when using the logarithmic transform approach where the four lognormal distributions all have the same mode in Gaussian space.

Table 21.1 The Difference Between the Lognormal Mode and Median as a Function of the Variance.

Statistic	$\sigma^2 = 0.01$	$\sigma^2 = 0.1$	$\sigma^2 = 0.5$	$\sigma^2 = 1$
Median	1	1	1	1
Mode	0.99	0.9048	0.6065	0.3679

lognormal mode. This could be interpreted as introducing a bias in the lognormal space relative to the maximum likelihood state. Another important feature to notice here is that the median is the same for all four distributions, as the plot in Fig. 21.2B indicates.

The reason that the logarithmic transform from lognormal space to Gaussian space introduces a bias relative to the lognormal maximum likelihood state is due to the fact that all the higher-order moments of the lognormal distribution are projected on to the zero higher-order moments of the Gaussian distribution in the transform. Therefore, the skewness information from the lognormal distribution does not go over to the Gaussian space and is thus lost, and as such that property of the original lognormal distribution cannot be incorporated in the inversion back to lognormal space.

However, the logarithmic transform has been used in many different forms; examples can be found in operational numerical weather prediction centers to constrain the moisture control variable from going negative. It has been trailed in the Navy Research Laboratory Atmospheric Variational Data Assimilation System [86], as well as in the Meteorological Service of Canada's middle atmosphere data assimilation system [338]. It is still in operation in the National Oceanic and Atmospheric Administration's National Environmental Satellite, Data and Information Services' Microwave Integrated Retrieval System [46].

The starting point in deriving the cost functions associated with the logarithmic transform is to assume that $\mathbf{X} \equiv \ln \mathbf{x}$. This implies that if $\mathbf{x} \sim LN\,(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then $\mathbf{X} \sim G\,(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Given that the new change

of variable is now assumed to be a Gaussian random variable, we simply substitute this random variable into a Gaussian cost function. Therefore, the 3D VAR cost function for the transformed random variable, where we are assuming a multivariate Gaussian distribution for the observational errors, which is the formulation in [46], is

$$J(\mathbf{X}) = \frac{1}{2}\left(\mathbf{X}^T - \mathbf{X}^b\right)^T \mathbf{B}^{-1}\left(\mathbf{X}^T - \mathbf{X}^b\right) + \frac{1}{2}\left(\mathbf{y} - \mathbf{h}(\mathbf{x})\right)^T \mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{h}(\mathbf{x})\right). \tag{21.17}$$

The first feature to note about (21.17) is that the observation operator is still acting on the untransformed variable. This means that when we calculate the Jacobian of the observational component with respect to $\mathbf{X}$, we have to apply the chain rule,

$$\frac{\partial J_o}{\partial \mathbf{X}} \equiv \frac{\partial J_o}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \mathbf{X}}. \tag{21.18}$$

As $\mathbf{X} \equiv \ln \mathbf{x}$, then the inverse transform is $\mathbf{x} \equiv \exp\{\mathbf{X}\}$, which means that the second derivative on the right-hand side of (21.18) is $\exp\{\mathbf{X}\}$. This implies that

$$\frac{\partial J_o}{\partial \mathbf{X}} \equiv \frac{\partial J_o}{\partial \mathbf{x}}\exp\{\mathbf{X}\}, \tag{21.19}$$

where $\exp\{\mathbf{X}\}$ is a diagonal matrix whose entries are the componentwise exponentials of $\mathbf{X}$.

Therefore, the Jacobian of (21.17) with respect to $\mathbf{X}^t$ is

$$\nabla J\left(\mathbf{X}^t\right) = \mathbf{B}^{-1}\left(\mathbf{X}^t - \mathbf{X}^b\right) - \exp\left\{\mathbf{X}^t\right\}\mathbf{H}^T\left(\mathbf{x}^t\right)\mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{h}\left(\mathbf{x}^t\right)\right). \tag{21.20}$$

An important feature to note about the transform approach is that the matrix $\exp\left\{\mathbf{X}^t\right\}$ is equivalent to $\mathbf{W}_b$ from the lognormal modal approach. With that in mind, we can now show that the nonlinear solution to (21.20) is

$$\mathbf{X}^t = \mathbf{X}^b + \mathbf{B}\exp\left\{\mathbf{X}^t\right\}^T\mathbf{H}^T\left(\mathbf{x}^t\right)\mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{h}\left(\mathbf{x}^t\right)\right). \tag{21.21}$$

We can clearly see that the offset terms associated with the mode are missing in (21.21), and as such the solution to (21.21) is an approximation to median of the multivariate lognormal distribution.

Finally, if we consider the matrix component of the Hessian of (21.17), we obtain

$$HESS\left(J\left(\mathbf{X}^t\right)\right) \equiv \mathbf{B}^{-1} + \exp\left\{\mathbf{X}^t\right\}^T\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\exp\left\{\mathbf{X}^t\right\}.$$

As we mentioned in Chapter 4, we do not live in a one-distribution-type-fits-all world and as such we need to be able to develop data assimilation systems that can minimize differently distributed errors simultaneously. That was the reason for the development of the mixed Gaussian-lognormal distribution. Therefore, we now move on to the mixed distribution full field 3D VAR development.

21.4 Mixed Gaussian-Lognormal 3D VAR

In Chapter 4 we presented the mixed Gaussian-lognormal distribution, from [136]. We showed that this distribution has an interesting property for the mode where we saw that the Gaussian component of the

mode became a function of the covariances between the Gaussian and the lognormal random variables. We now summarize the derivation of the cost function for this distribution from [136,137].

The starting point for the derivation of the cost function for a mixed Gaussian-lognormal distribution is the definition of the errors. We are going to assume that there are p_1 background errors that have a Gaussian distribution and that there are q_1 lognormally distributed background errors, where $p_1 + q_1 = N$. For the observational component we shall assume that there are p_2 Gaussian distributed errors and q_2 lognormally distributed errors, where $p_2 + q_2 = N_o$. Therefore, the definition of the mixed background and observational errors are

$$\boldsymbol{\varepsilon}^b_{mx} \equiv \begin{pmatrix} \mathbf{x}^t_{p_1} - \mathbf{x}^b_{p_1} \\ \mathbf{x}^t_{q_1} \\ \overline{\mathbf{x}^b_{q_1}} \end{pmatrix} \quad \boldsymbol{\varepsilon}^o_{mx} \equiv \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \mathbf{y}_{q_2} \\ \overline{\mathbf{h}_{q_2}\left(\mathbf{x}^t\right)} \end{pmatrix}. \tag{21.22}$$

We have made two important assumptions in the definitions presented above: (1) that the background distribution components are approximating the correct type of distributions but with possible different parameters; and (2) that the observation operator approximates the correct distribution of the observations, again possibly with different parameters.

The associated multivariate Gaussian-lognormal probability density function for the errors defined in (21.22) was presented in Chapter 4, but applying it to the error definitions above and taking the negative logarithm to create the maximum likelihood problem yields the following cost function:

$$\begin{aligned} J_{mx}(\mathbf{x}) = & \frac{1}{2}\begin{pmatrix} \mathbf{x}^t_{p_1} - \mathbf{x}^t_{p_1} \\ \ln \mathbf{x}^t_{q_1} - \ln \mathbf{x}^t_{q_1} \end{pmatrix}^T \begin{pmatrix} \mathbf{B}_{p_1 p_1} & \mathbf{B}_{p_1 q_1} \\ \mathbf{B}_{q_1 p_1} & \mathbf{B}_{q_1 q_1} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{x}^t_{p_1} - \mathbf{x}^t_{p_1} \\ \ln \mathbf{x}^t_{q_1} - \ln \mathbf{x}^t_{q_2} \end{pmatrix} \\ & + \left\langle \begin{pmatrix} \mathbf{x}^t_{p_1} - \mathbf{x}^t_{p_1} \\ \ln \mathbf{x}^t_{q_1} - \ln \mathbf{x}^t_{q_1} \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p_1} \\ \mathbf{1}_{q_1} \end{pmatrix} \right\rangle \\ & + \frac{1}{2}\begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \ln \mathbf{y}_{q_2} - \ln \mathbf{h}_{q_2}\left(\mathbf{x}^t\right) \end{pmatrix}^T \begin{pmatrix} \mathbf{R}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{R}_{q_2 q_2} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \ln \mathbf{y}_{q_2} - \ln \mathbf{h}_{q_2}\left(\mathbf{x}^t\right) \end{pmatrix} \\ & + \left\langle \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \ln \mathbf{y}_{q_2} - \ln \mathbf{h}_{q_2}\left(\mathbf{x}^t\right) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p_2} \\ \mathbf{1}_{q_2} \end{pmatrix} \right\rangle, \end{aligned} \tag{21.23}$$

where $\mathbf{B}_{p_1 p_1}$ represents the variance and covariances between the p_1 Gaussian background errors, $\mathbf{B}_{p_1 q_1}$ and $\mathbf{B}_{q_1 p_1}$ represent the covariances between the p_1 Gaussian background errors and the q_1 lognormally distributed background errors, and finally $\mathbf{B}_{q_1 q_1}$ represents the variances and the covariances of the q_1 lognormally distributed background errors. For the observational component we have assumed that both measurement and representative errors are uncorrelated. Therefore, the submatrices of $\mathbf{R}$ are diagonal matrices, where $\mathbf{R}_{p_2 p_2}$ represents the variances for the p_2 Gaussian distributed observational errors and $\mathbf{R}_{q_2 q_2}$ represents the variances for the q_2 lognormally distributed observational errors.

The next step in the process of implementing a mixed distribution-based full field 3D VAR system is to determine the Jacobian of (21.23) with respect to $\mathbf{x}^t = \begin{pmatrix} \mathbf{x}^t_{p_1} \\ \mathbf{x}^t_{q_1} \end{pmatrix}$, where we are assuming that the distributions of the background components are of the same distribution type as that of the true state.

With this in mind, the Jacobian of (21.23) is

$$\nabla J_{mx}\left(\mathbf{x}^t\right) = \begin{pmatrix} \mathbf{I}_{p_1 p_1} & \mathbf{0}_{p_1 q_1} \\ \mathbf{0}_{q_1 p_1} & \mathbf{W}_{b,q_1 q_1} \end{pmatrix}^{-T} \left[\begin{pmatrix} \mathbf{B}_{p_1 p_1} & \mathbf{B}_{p_1 q_1} \\ \mathbf{B}_{q_1 p_1} & \mathbf{B}_{q_1 q_1} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{x}_{p_1}^t - \mathbf{x}_{p_1}^t \\ \ln \mathbf{x}_{q_1}^t - \ln \mathbf{x}_{q_1}^t \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{p_1} \\ \mathbf{1}_{q_1} \end{pmatrix} \right]$$
$$- \begin{pmatrix} \mathbf{H}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{W}_{o,q_2 q_2} \mathbf{H}_{q_2 q_2} \end{pmatrix}^{-T} \left[\begin{pmatrix} \mathbf{R}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{R}_{q_2 q_2} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \ln \mathbf{y}_{q_2} - \ln \mathbf{h}_{q_2}\left(\mathbf{x}^t\right) \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{p_2} \\ \mathbf{1}_{q_2} \end{pmatrix} \right]. \tag{21.24}$$

It is quite clear that the solution when (21.24) is set to zero is the mode of the mixed distribution, where the Gaussian components are centered about their equivalent expected values and the row sums of the covariance between the Gaussian and lognormal background errors, and that the lognormal modal component is a function of the covariances between the lognormal components. This is important as it allows the lognormal component to affect the Gaussian random variables.

Finally, if we consider the Hessian of (21.23) with respect to $\mathbf{x}^t$, we obtain

$$HESS\left(J_{mx}\left(\mathbf{x}^t\right)\right) = \begin{pmatrix} \mathbf{I}_{p_1 p_1} & \mathbf{0}_{p_1 q_1} \\ \mathbf{0}_{q_1 p_1} & \mathbf{W}_{b,q_1 q_1} \end{pmatrix}^{-T} \begin{pmatrix} \mathbf{B}_{p_1 p_1} & \mathbf{B}_{p_1 q_1} \\ \mathbf{B}_{q_1 p_1} & \mathbf{B}_{q_1 q_1} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{I}_{p_1 p_1} & \mathbf{0}_{p_1 q_1} \\ \mathbf{0}_{q_1 p_1} & \mathbf{W}_{b,q_1 q_1} \end{pmatrix}^{-1}$$
$$+ \begin{pmatrix} \mathbf{H}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{W}_{o,q_2 q_2} \mathbf{H}_{q_2 q_2} \end{pmatrix}^{-T} \begin{pmatrix} \mathbf{R}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{R}_{q_2 q_2} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{H}_{p_2 p_2} & \mathbf{0}_{p_2 q_2} \\ \mathbf{0}_{q_2 p_2} & \mathbf{W}_{o,q_2 q_2} \mathbf{H}_{q_2 q_2} \end{pmatrix}^{-1}. \tag{21.25}$$

21.4.1 Experiments With the Lorenz 1963 Model

The results presented in this section are from [137]. It is not unusual for the Lorenz 1963 model to be used to test initial formulation of new data assimilation ideas. We presented this model in Chapter 13 and showed results using the modified Euler scheme. For us to consider using the Lorenz model for a lognormal or a mixed 3D VAR full field situation, we need evidence that there is possible a lognormal component, or at least a non-Gaussian component. To this extent we know that the z component is positive definite, which implies through the uniqueness theorem that it cannot have a Gaussian error that is optimal, but it is still not implied that the distribution for this variable is a lognormal.

To try to ascertain which distribution the z component is approximately following, we build a climatology through a series of histograms of the occurrence of specific range of values of z. In Fig. 21.3 we have a set of four histograms which represent an increasing length of numerical integration of the Lorenz 63 nonlinear system of ordinary differential equations, as well as the best Gaussian and the best lognormal distribution fit to the data.

In Fig. 21.3A we have the histogram of the z component after only 100 time steps. We can see that there is a slight skewness to the data, but that the best Gaussian and the best lognormal fits are still quite similar but slightly apart. In Fig. 21.3B we have the histogram of the z component after 1000 time steps. We can see that there appears to be a bimodal structure to this variable. The best Gaussian and lognormal approximations to the data are starting to separate. In Fig. 21.3C we have the histograms for 10,000 time steps, where we can now clearly see two modes appearing in the z component. The Gaussian and the lognormal approximations are separating further, but the lognormal approximation does appears to approximate the primary mode quite well. Finally, in Fig. 21.3D we have the histogram of the z component after 100,000 time steps. The two modes are now quite distinguishable and we can see that the lognormal approximation captures the primary mode quite well and only slightly underestimates

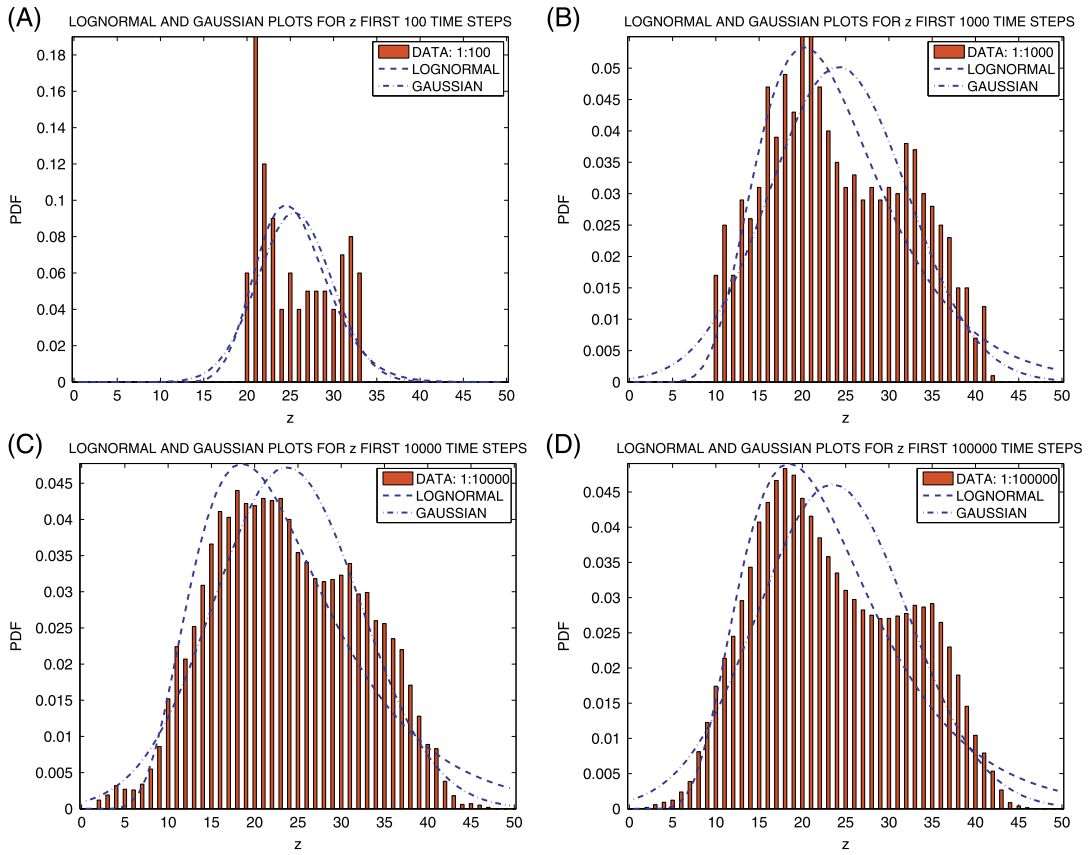


FIGURE 21.3

Climatology of the z component of the Lorenz 1963 model after a) 100, b) 1000, c) 10,000 and d) 100,000, time steps.

the secondary mode. However, the Gaussian approximation is torn between the two modes and wishes to be symmetric, and as such it is not capturing either mode; in fact it is assigning too high a probability to the transition area and underassigning probabilities to the two modes.

Given this quick climatological study of the Lorenz 1963 model, we would say that the z component of this model could be better approximated with a lognormal data assimilation system than a Gaussian-based one. We shall assume that the x and the y components of this model are better approximated with a Gaussian variational data assimilation scheme, therefore we shall use the mixed distribution-based approach.

The results that we present here, as we mentioned earlier, are from [137], where the mixed distribution approach was compared against the transform approach. The plots in Fig. 21.4 show two different scenarios. The first is when we have small observational errors, which implies the lognormal could be approximated by a Gaussian, and the second set of results are for when we have larger observational

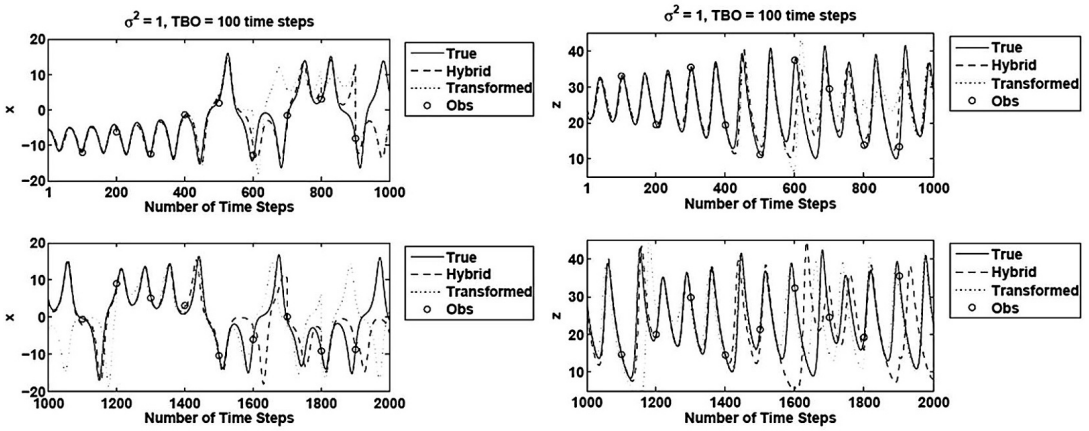


FIGURE 21.4

Figures from Fletcher and Zupanski (2007) from experiments comparing the mixed distribution approach with the logarithmic approach with the Lorenz 1963 model for full field 3D VAR.

errors and a longer time between assimilation updates. We see that for the first case there is not much difference between the two approaches. However, as the observational error variance increases and as the time between assimilation updates increase, we see that the mixed approach appears to be more stable than the transform approach. For more details about this study, see [137].

We have now presented the lognormal and the mixed Gaussian-lognormal formulations for three-dimensional variational data assimilation. The ease of these formulations comes from the ability to use Bayes' theorem with the definition or the error type and of the multivariate form of the probability density function. However, we now consider four-dimensional variational data assimilation where the derivations shown for the Gaussian-only formulation was based on a weighted least squares approach which is an estimator for the median of the analysis distribution.

21.5 Lognormal Calculus of Variation-Based 4D VAR

When we move to include the time component into the variational formulation of data assimilation, then as we saw for the Gaussian-based derivation of 4D VAR, the original approach from [251] was to introduce a weighted least squares type formulation by defining a functional of squared quantities that needed to be minimized. However, for the lognormal-based approach we have the problem in that we could define a weighted least square functional for the logarithmic transform approach, but this would only enable us to find the median solution through time with respect to the observational errors. As with the Gaussian functional formulation of 4D VAR, we require the functional to be in terms of the initial conditions for the model at the beginning of the assimilation window for a lognormal distribution-based approach.

There are two possible sets of approaches that could be taken to formulate a 4D VAR-based functional associated with lognormal, or a mixed lognormal-Gaussian distributed errors. The first formula-

tion is an adapted weighted least squares approach for the non-Gaussian case. The second approach is to define a functional whose minimum is equivalent to a lognormal, or a mixed distribution's mode. We consider the *near weighted least squares* approach first.

21.5.1 Near Weighted Least Squares Functional Formulation for Non-Gaussian 4D VAR

The starting point is to define the error structures that you wish to minimize. Recalling the definition of the background and observational errors for the Gaussian-based 4D VAR, we have

$$\boldsymbol{\varepsilon}_{b,0} \equiv \mathbf{x}_0^t - \mathbf{x}_0^b, \quad \text{and} \quad \boldsymbol{\varepsilon}_i^o \equiv \mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right).$$

Now if we were to form the change of variable $\mathbf{X}$, as $\mathbf{X} \equiv \ln \mathbf{x}$, then the error definition for the observational errors would stay unchanged, but the background error definition would be $\hat{\boldsymbol{\varepsilon}} \equiv \mathbf{X}_0^t - \mathbf{X}_0^b$. Thus the weighted least squares functional for the transform approach would be

$$J\left(\mathbf{X}^t\right) = \frac{1}{2}\left\langle \mathbf{B}_0^{-1}\left(\mathbf{X}_0^t - \mathbf{X}_0^b\right), \left(\mathbf{X}_0^t - \mathbf{X}_0^b\right)\right\rangle + \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right), \left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right)\right\rangle. \tag{21.26}$$

We can clearly see that the solution that minimizes the functional in (21.26) would have a structure similar to that of the median. This means that through the transform approach we are fitting the best median that minimizes the distance between the observations and their equivalent transform of the model variables through time.

Following a similar variational approach to that of the one presented for the Gaussian formulation of 4D VAR minimization problem, and including the adjoint properties presented earlier, the Jacobian of (21.26) can easily be shown to be

$$\nabla_{\mathbf{X}_{0t}} J\left(\mathbf{X}_o^t\right) \equiv B_0^{-1}\left(\mathbf{X}_0^t - \mathbf{X}_o^b\right) - \mathbf{W}_B^T \sum_{i=1}^{t_a} \mathbf{M}_{i,0}^T \mathbf{H}_i^T \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right), \tag{21.27}$$

where

$$\mathbf{W}_B^T \equiv \frac{\partial \mathbf{x}_0}{\partial \mathbf{X}_0}.$$

Therefore the nonlinear solution to (21.27), when set to zero, for $\mathbf{x}_0$, where we have inverted the logarithmic transform, is

$$\mathbf{x}_0 = \mathbf{x}_0^b \circ \exp\left\{\mathbf{B}_0 \mathbf{W}_B^T \sum_{i=1}^{t_a} \mathbf{M}_{0,i}^T \mathbf{H}_i^T \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)\right\}. \tag{21.28}$$

We can clearly see that the nonlinear solution in (21.28) is not in the form of a lognormal mode but nearer to a median.

If we had not introduced the logarithmic transform, then (21.26) would be equivalent to

$$\begin{aligned} J\left(\mathbf{x}_0\right) = {} & \frac{1}{2}\left\langle \mathbf{B}_0^{-1}\left(\ln \mathbf{x}_0 - \ln \mathbf{x}_{b,0}\right), \left(\ln \mathbf{x}_0 - \ln \mathbf{x}_{b,0}\right)\right\rangle \\ & + \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right), \left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)\right\rangle, \end{aligned} \tag{21.29}$$

where it can easily be shown that (21.28) is also the state that minimized (21.29).

If we now consider a weighted least squares approach where we have lognormally distributed observational errors, then the functional that has to be minimized for the observational component is

$$J_o = \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}\left(\mathbf{x}_0\right)\right)}, \ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}\left(\mathbf{x}_0\right)\right)}\right\rangle, \tag{21.30}$$

where the ratio in the logarithm is componentwise.

The minimum of (21.30) can be found by using the calculus of variation techniques introduced in Section 16.3, but we have to address the part of the functional in (21.30) that is a composite of four functions. To find the minimum of (21.30) we need the composite chain rule for four functions;

$$\left(f\left(g\left(h\left(i\left(\mathbf{x}\right)\right)\right)\right)\right)_{\mathbf{x}} = f_{\mathbf{x}}\left(g\left(h\left(i\left(\mathbf{x}\right)\right)\right)\right) g_{\mathbf{x}}\left(h\left(i\left(\mathbf{x}\right)\right)\right) h_{\mathbf{x}}\left(i\left(\mathbf{x}\right)\right) i_{\mathbf{x}}\left(\mathbf{x}\right). \tag{21.31}$$

The reason (21.31) is required is to be able to calculate the Taylor series expansion of $\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}\left(\mathbf{x}_0\right)\right)}$. Therefore the expressions for the functions in (21.31) are

$$\begin{aligned} f &= \ln\mathbf{x}, & f_{\mathbf{x}} &= \frac{1}{\mathbf{x}}, \\ g &= \frac{\mathbf{y}_i}{\mathbf{x}}, & g_{\mathbf{x}} &= -\frac{\mathbf{y}_i}{\mathbf{x}^2}, \\ h &= \mathbf{h}_i\left(\mathbf{x}\right), & h_{\mathbf{x}} &= \mathbf{H}_i, \\ i &= \boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right), & i_{\mathbf{x}} &= \mathbf{M}_i\left(\mathbf{x}_0\right), \end{aligned}$$

which results in

$$\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0+\delta\mathbf{x}_0\right)\right)} \approx \ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)} - \frac{1}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)}\mathbf{H}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)\mathbf{M}_i\left(\mathbf{x}_0\right)\delta\,\mathbf{x}_0, \tag{21.32}$$

where

$$\mathbf{W}_{o,i} \equiv \frac{\partial\ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)}{\partial\mathbf{x}_i}.$$

Through using the arguments from the Gaussian derivation of 4D VAR with respect to the properties of adjoint operators from Section 16.3, we can show that the Jacobian of the functional in (21.30) with respect to the logarithmic transformed variable $\mathbf{X}_0$ is

$$\nabla_{\mathbf{X}_0} J_o = \mathbf{W}_B^T\sum_{i=1}^{t_a}\mathbf{M}_i^T\mathbf{H}_i^T\mathbf{W}_{o,i}^T\mathbf{R}_i^{-1}\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)}. \tag{21.33}$$

If we now consider the solution to the weighted least squares functional for a logarithmic transform combined with the lognormal observational weighted least squares functional, then the nonlinear solution that minimizes the combination of (21.33) and (21.28) when set to zero is

$$\mathbf{x}_0 = \mathbf{x}_0^b \circ \exp\left\{\mathbf{B}_0\mathbf{W}_B^T\sum_{i=1}^{t_a}\mathbf{M}_i^T\mathbf{H}_i^T\mathbf{W}_{o,i}^T\,\mathbf{R}_i^{-1}\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_i\left(\mathbf{x}_0\right)\right)}\right\}. \tag{21.34}$$

The nonlinear solution presented in (21.34) does appear to be combining the median state of the background error distribution with the median state of the observational error distribution, which is what we would expect both from the logarithmic transform approach as well as the basis of the functional being a weighted least squares problem.

By using the same arguments just presented, it is possible to show that if we defined the observational errors to follow a mixed Gaussian-lognormal distribution, for a 4D VAR system as

$$\boldsymbol{\varepsilon}_i^{o,mx} \equiv \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\boldsymbol{\mathcal{M}}\left(\mathbf{x}_0\right)\right) \\ \ln \frac{\mathbf{y}_{q_2}}{\mathbf{h}_{q_2}\left(\boldsymbol{\mathcal{M}}(\mathbf{x}_0)\right)} \end{pmatrix}, \tag{21.35}$$

then the state that minimized the associated functional would be approximate to a mixed distribution median (see [129] for more details).

Therefore, if we extend the weighted least squares approach of Gaussian based 4D VAR to lognormally, and mixed Gaussian-lognormal, distributed errors, then we obtain estimates for the median of these distributions. However, in [129] it is shown that is possible to define a functional such that the minimum appears to be a mode of a lognormal, or a mixed Gaussian-lognormal, distribution.

21.5.2 Functional Form of a Modal Approach for Non-Gaussian Distribution-Based 4D VAR

We have seen with our dealings with the lognormal distribution, that the mode is a function of the median, and as such, given that we have functional forms for the median of the lognormal distribution, and the mixed Gaussian-lognormal, distributions, we should be able to adapt these so that the state that minimizes the functional is approximately the mode of a lognormal/mixed distribution. What we require is an offset term to the median, so that the solution of the Jacobian of the functional contains the row sum of the covariance matrix. To achieve this aim, we consider the following functional:

$$\begin{aligned} J\left(\mathbf{x}\right) = &\frac{1}{2}\left\langle \mathbf{B}_0^{-1}\left(\ln \mathbf{x}_0 - \ln \mathbf{x}_0^b\right) + \mathbf{1}, \left(\ln \mathbf{x}_0 - \ln \mathbf{x}_{b,0}\right)\right\rangle \\ &+ \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right), \left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)\right\rangle. \end{aligned} \tag{21.36}$$

Through using the techniques outlined earlier for the first variation of a functional, combined with the properties of adjoints, we can easily show that the gradient of (21.36) is

$$\nabla J = \mathbf{W}_{B,0}^{-T}\left(\mathbf{B}_0^{-1}\left(\ln \mathbf{x}_0 - \ln \mathbf{x}_{b,0}\right) + \mathbf{1}\right) - \sum_{i=1}^{t_a}\mathbf{M}_i^T\mathbf{H}_i^T\ \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right). \tag{21.37}$$

Setting (21.37) equal to zero, and rearranging, results in the nonlinear solution

$$\mathbf{x}_0 = \mathbf{x}_0^b \circ \exp\left\{-\mathbf{B}_0\mathbf{1}\right\}\exp\left\{\mathbf{B}_0\mathbf{W}_B^T\sum_{i=1}^{t_a}\mathbf{M}_{0,i}^T\mathbf{H}_i^T\mathbf{R}_i^{-1}\left(\mathbf{y}_i - \ \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)\right\}. \tag{21.38}$$

We can clearly see that we have been able to obtain the row sum of the background error covariance matrix, which we needed for our solution to be consistent with the structure of the mode of a multivariate lognormal distribution.

In [129] it is shown that it is possible to define a functional for lognormally distributed observational errors such that the minimum of that functional is the mode of a lognormal distribution, which is

$$J_o = \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\left(\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)} + \mathbf{R}_i\mathbf{1}_{N_i}\right), \ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)}\right\rangle, \tag{21.39}$$

where $\mathbf{1}_{N_i}$ is a vector of 1s of dimension $N_i \times 1$, where N_i is the number of observations at time $t = t_i$.

Following the first variation arguments as in the derivations of the Jacobian of the other functional in this section, and using adjoint properties, results in the Jacobian of (21.39) as

$$\nabla_{\mathbf{x}_0} = -\sum_{i=1}^{t_a}\mathbf{M}_{0,i}^T\mathbf{H}_i^T\mathbf{W}_{o,i}^T\mathbf{R}_i^{-1}\left(\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)} + \mathbf{R}_i\mathbf{1}_i\right). \tag{21.40}$$

The nonlinear solution of (21.40), when combined with the lognormal background component of (21.36), is

$$\mathbf{x}_0 = \mathbf{x}_0^b \exp\left\{\mathbf{B}_0\mathbf{1}\right\}\exp\left\{-\mathbf{B}_0\mathbf{W}_B^T\sum_{i=1}^{t_a}\mathbf{M}_{0,i}^T\,\mathbf{H}_i^T\mathbf{W}_{o,i}^T\mathbf{R}_i^{-1}\left(\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)} + \mathbf{R}_i\mathbf{1}_i\right)\right\}, \tag{21.41}$$

which is the mode, as detailed in [137], but extended to all of the innovations throughout the assimilation window. That is to say that the solution is finding the most likely state that minimizes the distance between the observations and their model equivalent.

Given the functional form for the mode of a lognormal-based functional state above, it is possible to easily extend this definition to a mixed Gaussian-lognormal distribution to describe the background error as

$$J_B = \frac{1}{2}\left\langle \mathbf{B}_0^{-1}\begin{pmatrix}\mathbf{x}_{p_1,0}^t - \mathbf{x}_{p_1,0}^b \\ \ln\mathbf{x}_{q_1,0}^t - \ln\mathbf{x}_{q_1,0}^b\end{pmatrix} + \begin{pmatrix}\mathbf{0}_{p_1} \\ \mathbf{1}_{q_1}\end{pmatrix}, \begin{pmatrix}\mathbf{x}_{p_1,0}^t - \mathbf{x}_{p_1,0}^b \\ \ln\mathbf{x}_{q_1,0}^t - \ln\mathbf{x}_{q_1,0}^b\end{pmatrix}\right\rangle. \tag{21.42}$$

The functional for the mode of mixed Gaussian-lognormal observational error distribution can easily be verified to be

$$J_o = \frac{1}{2}\sum_{i=1}^{t_a}\left\langle \mathbf{R}_i^{-1}\left(\ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)} + \mathbf{R}_i\begin{pmatrix}\mathbf{0}_{p_2,i} \\ \mathbf{1}_{q_2,i}\end{pmatrix}\right), \ln\frac{\mathbf{y}_i}{\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}\left(\mathbf{x}_0\right)\right)}\right\rangle. \tag{21.43}$$

Therefore, combining (21.42) and (21.43) and forming the first variation, which is the equivalent the Jacobian of the cost function, it can be shown that the nonlinear solution is equivalent to (21.41) with **1**s of different dimensions replaced with the equivalent vector for the mixed formulation. This means that if we minimize (21.42) and (21.43), then we fit the mode as the best estimate to minimize difference between observations and the model equivalent for both the Gaussian and lognormal distributed errors.

Note. In [129] the authors present the functional form for the mean of a lognormal distribution; however, the functional is a componentwise approximation to the mean because as we have already seen the multivariate mean is defined componentwise for all distributions.

However, if we need to quickly change distributions, it is not always obvious, or easy, to rewrite the multivariate distribution into a functional form to perform calculus of variation upon to find the

minimum. This was the motivation for the work in [129] where an alternative approach involving probability theory and Bayes' theorem was shown to be equivalent to the functional forms shown for all of the distributions that we have considered for 4D VAR but where **NO** assumptions on which PDF is being used.

21.6 Bayesian-Based 4D VAR

The motivation to develop a Bayesian-based model for 4D VAR that is similar to that for 3D VAR was to be able to substitute different multivariate PDFs for the errors when they occur into a set equation so that we could more realistically model the distribution of the error. Through having a Bayesian model with respect to time for the observations means that if we have a set of gamma-distributed observational errors at some time inside the assimilation windows, along with lognormally and possibly Gaussian at some other times, then we would take the Bayesian model for that window and substitute the relevant distribution at the associated time.

While at the time of writing first edition of this textbook there did not exist a variational, or an ensemble, data assimilation solely based upon a multivariate gamma distribution, since then there has been the gamma-inverse gamma filter, and we will present this filer later. However, we need to start setting the mathematical/statistical/probabilistic infrastructure in place to be able to in the future for any distribution. We should note that when mentioning non-Gaussian distributions, there are particle filters and Monte Carlo based methods that can handle non-Gaussian distributions and may be able to minimize gamma-distributed errors.

Before we derive the Bayesian model for 4D VAR, we introduce the concepts of **Bayesian networks**.

21.6.1 Bayesian Networks

In this section two different probability frameworks for 4D VAR, which are based upon a technique in probability theory called Bayesian networks [331], are derived.

Bayesian networks are graphical methods which, according to [331], have three roles:

1. to provide convenient means of expressing substantive assumptions;
2. to facilitate economical representation of joint probability functions; and
3. to facilitate efficient inference from observations.

In data assimilation, all of the properties above are desirable in describing how the errors are behaving with respect to the information that is at hand.

The main feature of Bayesian networks is that they enable the removal of non-dependent probability events from the multiple probabilistic event version of Bayes' theorem. The process of the removal of the non-dependent events has many names but the two most common are **conditional independence** and **Markov processes**.

Bayesian networks are a type of graphical model that consist of nodes, arrows between nodes, and probability assignments. These are also referred to as directed graphs. If we have an arrow pointing from node X to node Y, then X is said to be the parent of Y. A node without a parent is referred to

as a root node. Nodes can have ancestors and descendants. That is X is an ancestor of Y and Y is a descendant of X.

In Fig. 21.5 we have an example of a Bayesian network and is called a **directed acyclic graph (DAG)**. The different letters (nodes) can represent random variables that are either continuous, discrete or both [331]. Given our definitions above then we can see that node A is a root node but is also a parent node of B and D. Node D is the parent of nodes G and E while for node H we have nodes G, F, and E as its parent nodes.

We can also say that node A is an ancestor of all the nodes in the graph, while all the nodes are descendants of node A. We also have that nodes B and D are ancestors of node E, while nodes F and H are descendants of E. Given these brief example of a Bayesian network, we move on to define conditional independence, which will play an important role later in this chapter, as follows.

Definition 21.1. For every variable X in the DAG, and every set $\mathbf{Y}$ of variables such that it does not include the set $\mathbf{DE}(X)$ of descendent's of X then X is **conditionally independent** from $\mathbf{Y}$, given the set $\mathbf{PA}(X)$ of its parents and therefore

$$P(X|\mathbf{PA}(X), \mathbf{Y}) = P(X|\mathbf{PA}(X)), \tag{21.44}$$

The advantage of Bayesian networks is that they allow the simplification of large joint distributions, $P(x_1, x_2, \ldots, x_N)$, where there are 2^N outcomes that have to be considered, through highlighting the events that are connected to each other.

The multiple event version of Bayes' theorem allows the multivariate joint probability to be written in terms of the product of conditional distributions and a marginal distribution, given by

$$P(X_N, X_{N-1}, \ldots, X_2, X_1) = \left(\prod_{i=2}^{N} P\left(X_i \mid \hat{X}_{i-1}\right)\right) P(X_1), \tag{21.45}$$

where $\hat{X}_{i-1} \equiv X_{i-1}, X_{i-2}, \ldots, X_1$.

It is clear from (21.45) that there are many expressions to consider in the joint probability function, and this is where Bayesian networks become useful. As a way to demonstrate this, we consider Fig. 21.6, which is the DAG for strong constraint 4D VAR for a small sample. The joint probability for

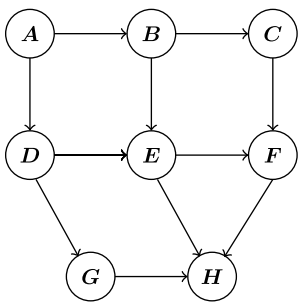


FIGURE 21.5

A toy example of a Direct Acyclic Graph (DAG).

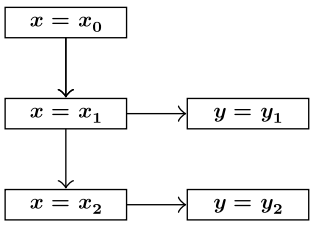


FIGURE 21.6

The DAG for a small size strong constraint 4D VAR.

Fig. 21.6 is

$$P(X_0, X_1, Y_1, X_2, Y_2) = P(X_0)\, P(X_1 \mid X_0)\, P(Y_1 \mid X_1, X_0)\, P(X_2 \mid Y_1, X_1, X_0) \times P(Y_2 \mid X_2, Y_1, X_1, X_0), \tag{21.46}$$

where X_i, $i = 0, 1, 2$ is the random event that $\mathbf{x}_i = \mathbf{x}_i^t$, and Y_i, $i = 1, 2$ is the random event that $\mathbf{y}_i = \mathbf{y}_i^t = \mathbf{x}_i$ is the model state and $\mathbf{y}_i$ is a set of observations at $t = t_i$. This is where Bayesian networks are useful as they identify non-dependencies in (21.46) to be removed. This is accomplished through the definition of Markov Parents/conditional independence:

Definition 21.2 (Markovian Parents). For every variable X in the DAG, and every set $\mathbf{Y}$ of variables such that it does not include the set $\mathbf{DE}(X)$ of descendants of X, then X is conditionally independent from $\mathbf{Y}$ given the set $\mathbf{PA}(X)$ of its parents and therefore [331]

$$P(X \mid \mathbf{PA}(X), \mathbf{Y}) = P(X \mid \mathbf{PA}(X)). \tag{21.47}$$

Mathematically, Definition 21.2 can be written as

$$P(X_1, \ldots, X_n) = \left[\prod_{i=2}^{n} P(X_i \mid \mathbf{PA}(X_i))\right] P(X_1), \tag{21.48}$$

where (21.48) can be interpreted as the **conditional independent version of Bayes' theorem**.

Therefore, we need to identify the Markov parents in Fig. 21.6. First we see that X_0 is the root node. For the second event, X_1, then the parent node is X_0. For the first set of observations the parent node is X_1, not the set $\{X_1, X_0\}$. Therefore the joint probability is $P(Y_1 \mid X_1)$. However, when we consider X_2 it is clear that this is dependent on the events X_1 and X_0. By assuming a hidden Markov model, the conditional probability of event X_2 becomes $P(X_2 \mid X_1)$.

Combining all the arguments above results in the final joint probability as

$$P(X_0, X_1, Y_1, X_2, Y_2) = P(X_0)\, P(X_1 \mid X_0)\, P(Y_1 \mid X_1)\, P(X_2 \mid X_1)\, P(Y_2 \mid X_2,). \tag{21.49}$$

There is one more simplification that can be made to (21.46), which arises from the **perfect model assumption**. This assumption implies that if the initial conditions are the true initial conditions then all of the proceeding model states are true. Therefore, the conditional probability of the future model

states is 1. Using this information results in the joint probability, or posterior, distribution for strong constraint 4D VAR for three model states as

$$P\left(X_0, X_1, Y_1, X_2, Y_2\right) = P\left(X_0\right) P\left(Y_1 \mid X_1\right) P\left(Y_2 \mid X_2\right). \tag{21.50}$$

The arguments used to derive (21.50) can be extended to t_a sets of observations, which results in

$$P\left(X_0, X_1, Y_1, \ldots, X_{N_o}, Y_{N_o}\right) = P\left(X_0\right) \prod_{i=1}^{t_a} P\left(Y_i \mid X_i\right). \tag{21.51}$$

However, (21.51) is an expression for the joint probability density function, but for the variational-based data assimilation systems we require an expression for the mode of this PDF. We therefore follow the same argument for the derivation of the mode of the posterior distribution for 3D VAR, where we follow the log-likelihood approach. Thus the mode of (21.51) is the minimum of the cost function

$$J\left(\mathbf{x}\right) = -\ln P\left(X_0\right) - \sum_{i=1}^{N_o} \ln P\left(Y_i \mid X_i\right). \tag{21.52}$$

We should note here that we have made no assumptions about which distributions the PDFs in (21.52) represent. Therefore, (21.52) can have any distribution substituted in at any time, which would give us a cost function to minimized that is a function of multiple different types of PDFs in time, which was the motivation to derive such a general probabilistic formulation for 4D VAR in [129].

21.6.2 Equivalence of the Weighted Least Squares and Probability Models for Multivariate Gaussian Errors

To show the equivalence between the weighted least squares approach and the probability model that has just been derived, for the multivariate Gaussian case, we start as always with the definition for multivariate Gaussian background and observational errors:

$$\begin{aligned} \boldsymbol{\varepsilon}_0^b &= \mathbf{x}_0^t - \mathbf{x}_0^b, \qquad \boldsymbol{\varepsilon}_i^o = \mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right), \\ \boldsymbol{\varepsilon}_0^b &\sim G\left(\mathbf{0}, \mathbf{B}\right), \qquad \boldsymbol{\varepsilon}_i^o \sim G\left(\mathbf{0}, \mathbf{R}_i\right), \end{aligned}$$

where G stands for the multivariate Gaussian distribution, which we have shown is defined as

$$G\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = (2\pi)^{-\frac{N}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}\left(\mathbf{x} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\mathbf{x} - \boldsymbol{\mu}\right)\right\}, \tag{21.53}$$

where $\boldsymbol{\Sigma}$ is a covariance matrix and $\boldsymbol{\mu}$ is the vector of the expectations of the components of the random vector, $\mathbf{x}$.

Combining the definitions of the background and observational errors with (21.53) and substituting into (21.52) results in

$$J\left(\mathbf{x}\right) = \frac{1}{2}\left(\mathbf{x}_0^t - \mathbf{x}_0^b\right)^T \mathbf{B}_0^{-1}\left(\mathbf{x}_0^t - \mathbf{x}_0^b\right) + \frac{1}{2}\sum_{i=1}^{t_a}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)^T \mathbf{R}_i^{-1}\left(\mathbf{y}_i - \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right),$$

which is the same as the 4D VAR cost function that we derived from the weighted least squares approach in Section 16.3.

21.6.3 Equivalence of the Lognormal Functional Approach

We have just proven that for Gaussian distributed background and observational errors, the Bayesian network-based generalized maximum likelihood formulation in (21.52) is equivalent to the weighted least squares approach for 4D VAR. We now show that this is also true for lognormally distributed background and observational errors.

We recall the definitions of the lognormally distributed background and observational errors

$$\boldsymbol{\varepsilon}_0^b = \frac{\mathbf{x}_0^t}{\mathbf{x}_0^b}, \qquad \boldsymbol{\varepsilon}_i^o = \frac{\mathbf{y}_i}{\mathbf{h}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)},$$
$$\boldsymbol{\varepsilon}_0^b \sim MLN\left(\mathbf{0}, \mathbf{B}\right), \quad \boldsymbol{\varepsilon}_i^o \sim MLN\left(\mathbf{0}, \mathbf{R}_i\right),$$

where MLN stands for multivariate lognormal, where the definition for this distribution is

$$f\left(\mathbf{x}\right) = (2\pi)^{-\frac{N}{2}} \left|\boldsymbol{\Sigma}\right|^{-1} \prod_{i=1}^{N} \left(\frac{1}{x_i}\right) \exp\left\{-\frac{1}{2}\left(\ln\,\mathbf{x} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma}^{-1}\left(\ln \mathbf{x} - \boldsymbol{\mu}\right)\right\}.$$

As a reminder, $\boldsymbol{\mu} = E\left(\ln \mathbf{x}\right)$ and $\boldsymbol{\Sigma}$ is the covariance matrix for $\ln \mathbf{x}$ **not** $\mathbf{x}$.

Substituting the information above into (21.52) results in

$$\begin{aligned} J\left(\mathbf{x}\right) = {} & \frac{1}{2}\left(\ln \mathbf{x}_0^t - \ln \mathbf{x}_0^b\right)^T \mathbf{B}^{-1}\left(\ln \mathbf{x}_0^t - \ln\,\mathbf{x}_0^b\right) + \left\langle\left(\ln \mathbf{x}_0^t - \ln \mathbf{x}_0^b\right), \mathbf{1}_N\right\rangle \\ & + \frac{1}{2}\sum_{k=1}^{t_a}\left(\ln \mathbf{y}_i - \ln \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right)^T \mathbf{R}_i^{-1}\left(\ln \mathbf{y}_i - \ln \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right) \\ & + \sum_{i=1}^{t_a}\left\langle\left(\ln \mathbf{y}_i - \ln \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right)\right), \mathbf{1}_{N_i}\right\rangle, \end{aligned} \tag{21.54}$$

which is equivalent to the cost function that was derived from the modal-based functional form.

Remark 21.3. However, we should note here that if we were to use the transform approach we would follow the Gaussian derivation from (21.52), but given that the property that the logarithmic transform results in an approximation of the median of the multivariate lognormal analysis distribution, it is possible to take the general joint probability expression in (21.51) and instead of seeking the maximum likelihood state, we could easily define the unbiased state, which would be

$$\mathbf{x}_{med}^a \equiv \int_{x_{l_1}}^{x_1^a}\int_{x_{l_2}}^{x_2^a}\cdots\int_{x_{l_N}}^{x_n^a}\left(P\left(X_0\right)\prod_{i=1}^{t_a}P\left(Y_i \mid X_i\right)\right)dx_1 dx_2 \ldots d_{x_N} = \frac{1}{2}, \tag{21.55}$$

where x_{l_i} is the lower limit of the probability density function, again we are not assuming any specific distribution. If we were to consider the minimum variance estimator of (21.51), then for each component of the multivariate analysis mean we would have to solve

$$x_{mean}^a = \mathbb{E}\left[x_i\right] = \int_{x_{l_i}}^{x_{u_i}} x_i^a\left[P\left(X_0\right)\prod_{i=1}^{t_a}P\left(Y_i \mid X_i\right)\right]_i dx_i^a, \tag{21.56}$$

where x_{u_i} is the upper limit of the distribution, and where the joint probability from (21.51) has to be valued as a univariate distribution for that specific component of the analysis mean.

21.6.4 Mixed Distribution Equivalency to Weighted Least Squares Approach

As we have mentioned, there have been no assumptions made upon which distributions can and cannot be used with (21.52) when seeking a maximum likelihood state. Therefore, if we have the situation where we have mixed Gaussian-lognormal distributed background and observational errors,

$$\boldsymbol{\varepsilon}_0^b \equiv \begin{pmatrix} \mathbf{x}_{p_1}^t - \mathbf{x}_{p_1}^b \\ \mathbf{x}_{q_1}^t \\ \overline{\mathbf{x}_{q_1}^b} \end{pmatrix}, \quad \boldsymbol{\varepsilon}_i^o \equiv \begin{pmatrix} \mathbf{y}_{p_2} - \mathbf{h}_{p_2}\left(\mathbf{x}^t\right) \\ \mathbf{y}_{q_2} \\ \overline{\mathbf{h}_{q_2}\left(\mathbf{x}^t\right)}. \end{pmatrix},$$
$$\boldsymbol{\varepsilon}_0^b \sim MX\left(\mathbf{0}, \mathbf{B}\right), \qquad \boldsymbol{\varepsilon}_i^o \sim MX\left(\mathbf{0}, \mathbf{R}_i\right),$$

where the multivariate mixed Gaussian-lognormal distribution is defined as

$$MX\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = (2\pi)^{-\frac{N}{2}} \left|\boldsymbol{\Sigma}\right|^{-1} \prod_{i=p+1}^{N} \left(\frac{1}{x_i}\right) \exp\left\{ \begin{pmatrix} \mathbf{x}_p - \boldsymbol{\mu}_p \\ \ln \mathbf{x}_q - \boldsymbol{\mu}_q \end{pmatrix}^T \boldsymbol{\Sigma}^{-1} \begin{pmatrix} \mathbf{x}_p - \boldsymbol{\mu}_p \\ \ln \mathbf{x}_q - \boldsymbol{\mu}_q \end{pmatrix} \right\},$$

then we would obtain the following cost function from (21.52):

$$\begin{aligned} J\left(\mathbf{x}\right) = {} & \frac{1}{2} \begin{pmatrix} \mathbf{x}_{p_1}^t - \mathbf{x}_{p_1}^b \\ \ln \mathbf{x}_{q_1}^t - \ln\, \mathbf{x}_{q_1}^t \end{pmatrix}^T \mathbf{B}_0^{-1} \begin{pmatrix} \mathbf{x}_{p_1}^t - \mathbf{x}_{p_1}^b \\ \ln \mathbf{x}_{q_1}^t - \ln\, \mathbf{x}_{q_1}^t \end{pmatrix} + \left\langle \begin{pmatrix} \mathbf{x}_{p_1}^t - \mathbf{x}_{p_1}^b \\ \ln \mathbf{x}_{q_1}^t - \ln\, \mathbf{x}_{q_1}^t \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p_1} \\ \mathbf{1}_{q_1} \end{pmatrix} \right\rangle \\ & + \frac{1}{2} \sum_{i=1}^{t_a} \begin{pmatrix} \mathbf{y}_{p_{2,i}} - \mathbf{h}_{p_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}\right)\right) \\ \ln \mathbf{y}_{q_{2,i}} - \ln \mathbf{h}_{q_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right) \end{pmatrix}^T \mathbf{R}_i^{-1} \begin{pmatrix} \mathbf{y}_{p_{2,i}} - \mathbf{h}_{p_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right) \\ \ln \mathbf{y}_{q_{2,i}} - \ln\, \mathbf{h}_{q_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}\right)\right) \end{pmatrix}, \\ & + \sum_{i=1}^{t_a} \left\langle \begin{pmatrix} \mathbf{y}_{p_{2,i}} - \mathbf{h}_{p_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0\right)\right) \\ \ln \mathbf{y}_{q_{2,i}} - \ln \mathbf{h}_{q_{2,i}}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}\right)\right) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p_{2,i}} \\ \mathbf{1}_{q_{2,i}} \end{pmatrix} \right\rangle. \end{aligned} \tag{21.57}$$

21.7 Bayesian Networks Formulation of Weak Constraint/Model Error 4D VAR

The weak constraint, or attempting to account for model error, approach, as outlined in [389], is often considered the more realistic approach to data assimilation, as it is well known that the versions of the governing equations used in the numerical weather prediction systems are not exact. Many simplifications are made through scale analysis to the continuous equations [332], which then enables the discrete approximations to be simpler. This is also true of many of the geophysical sciences where the true governing equations are highly nonlinear to solve analytically or numerically. While this is a type of model error, it is not the only form of the error that can affect data assimilation systems. As we saw with the Eady model, it is possible that the numerical approximation to the continuous equations can introduce further errors into the approximation of the nonlinear geophysics.

Model errors are a major problem for data assimilation methods, as the strong constraint approach assumes that the numerical model is perfect. However, in [85] the model error is shown to be a significant term. In this section we consider a first-order Markov variable (chain) approach as an approximation for the model error term; this is equivalent to assuming that the current model state is

only conditionally dependent on the model state at the previous model time. The DAG for model error approach is presented in Fig. 21.7.

Therefore, the conditional independent version of Bayes' theorem for Fig. 21.7 is

$$P(X_0, X_1, X_2, Y_1, X_3, X_4, Y_2) = P(X_0)\, P(X_1 \mid X_0)\, P(X_2 \mid X_1)\, P(Y_1 \mid X_2)\, P(X_3 \mid X_2)\, P(X_4 \mid X_3)\, P(Y_2 \mid X_4), \tag{21.58}$$

which generalizes to

$$P\left(X_0, X_1, X_2, Y_1, \ldots, X_{N-1}, Y_{N_o}, X_N\right) = P(X_0) \prod_{i=1}^{N} P\left(X_i \left| X_{i-1}\right.\right) \prod_{j=1}^{t_a} P\left(Y_j \left| X_i\right.\right), \tag{21.59}$$

where N is the total number of model evaluations in the assimilation window. Note we cannot assume that the probability of the model state given that the initial conditions are correct is equal to 1 as we could with the strong constraint approach, but we have been able to eliminate the dependence on all the early model states except the model value at the previous time step.

As with the strong constraint formulations, if we are considering a variational-based four-dimensional system, then it is the maximum likelihood state that is required; therefore, using the log-likelihood method, the cost function is

$$J(\mathbf{x}) = -\ln P(X_0) - \sum_{i=1}^{N} \ln P(Y_i \mid X_i) - \sum_{i=1}^{N_o} \ln P\left(X_i \left| X_{i-1}\right.\right). \tag{21.60}$$

The derivation just presented results in an equivalent expression as that of the state augmentation approach from [503] and which we presented in Chapter 16.

As with the strong constraint approach, it is possible to define the analysis median and the analysis mean for the model error generalized probability model. For the analysis median would find the value of $\mathbf{x}^a$ such that the cumulative density function of (21.59) is equal to 0.5. To find each component of the analysis mean we would have to apply the expectation operator to (21.59).

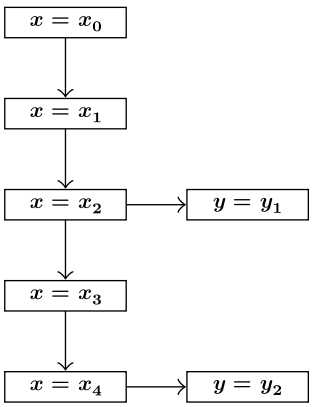


FIGURE 21.7

The DAG for a small sample weak constraint 4D VAR.

We now derive the weak constraint cost function for Gaussian, lognormal, and mixed distributed model errors.

Gaussian model errors

If the model errors are assumed to be Gaussian distributed, then they could be defined as

$$\boldsymbol{\varepsilon}_m = \mathbf{x}_i - \boldsymbol{\mathcal{M}}_{i-1,i}\left(\mathbf{x}_{b,i-1}\right). \tag{21.61}$$

Therefore, using the Gaussian error definitions for the background and observational errors with (21.61) and substituting into (21.60) results in

$$\begin{aligned} J\left(\mathbf{x}_0^t\right) = &\frac{1}{2}\left(\mathbf{x}_0^t - \mathbf{x}_0^b\right)^T \mathbf{B}^{-1}\left(\mathbf{x}_0 - \mathbf{x}_0^b\right) + \frac{1}{2}\sum_{j=1}^{t_a}\left(\mathbf{y}_j - \mathbf{h}_j\left(\boldsymbol{\mathcal{M}}_{0,j}\left(\mathbf{x}_0\right)\right)\right)^T \mathbf{R}_j^{-1}\left(\mathbf{y}_j - \mathbf{h}_j\left(\boldsymbol{\mathcal{M}}_{0,j}\left(\mathbf{x}_0\right)\right)\right) \\ &+ \sum_{i=1}^{N}\left(\mathbf{x}_i - \boldsymbol{\mathcal{M}}_{i-1,i}\left(\mathbf{x}_{i-1}^b\right)\right)^T \mathbf{Q}_i^{-1}\left(\mathbf{x}_i - \boldsymbol{\mathcal{M}}_{i-1,i}\left(\mathbf{x}_{i-1}^b\right)\right), \end{aligned} \tag{21.62}$$

which is the same as the expression in [450], where $\mathbf{Q}_i$ is the model error covariance matrix at time t_i.

Lognormally distributed model errors

In [129] the weak constraint, model error, 4D VAR equations were derived for both lognormally, and mixed Gaussian-lognormal, distributed model errors. The definition for lognormally distributed model errors is

$$\boldsymbol{\varepsilon}_k^m = \frac{\mathbf{x}_k^t}{\mathbf{x}_k^b}, \quad k = 1, 2, \ldots, K, \tag{21.63}$$

where $\mathbf{x}_k^b = \boldsymbol{\mathcal{M}}_{k-1,k}\left(\mathbf{x}_{k-1}^b\right)$ and K is the total number of time steps in the assimilation window. Therefore, substituting this information, combined with the definitions for lognormally distributed background and observational errors in (21.60), results in

$$\begin{aligned} J\left(\mathbf{x}_0^t\right) = &\frac{1}{2}\left(\ln\mathbf{x}_0^t - \ln\mathbf{x}_0^b\right)^T \mathbf{B}_0^{-1}\left(\ln\mathbf{x}_0^t - \ln\mathbf{x}_0^b\right) + \left\langle\left(\ln\mathbf{x}_0^t - \ln\mathbf{x}_0^b\right), \mathbf{1}_N\right\rangle \\ &+ \frac{1}{2}\sum_{i=1}^{N_o}\left(\ln\mathbf{y}_i - \ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right)^T \mathbf{R}_i^{-1}\left(\ln\mathbf{y}_i - \ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right) \\ &+ \frac{1}{2}\sum_{k=1}^{K}\left(\ln\mathbf{x}_k - \ln\boldsymbol{\mathcal{M}}_{k-1,k}\left(\mathbf{x}_{k-1}^b\right)\right)^T \mathbf{Q}_k^{-1}\left(\ln\mathbf{x}_k - \ln\boldsymbol{\mathcal{M}}_{k-1,k}\left(\mathbf{x}_{k-1}^b\right)\right) \\ &+ \sum_{i=1}^{N_o}\left\langle\left(\ln\mathbf{y}_i - \ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_0^t\right)\right)\right), \mathbf{1}_{N_{o,i}}\right\rangle + \sum_{k=1}^{K}\left\langle\left(\ln\mathbf{x}_k - \ln\boldsymbol{\mathcal{M}}_{k-1,k}\left(\mathbf{x}_{k-1}^b\right)\right), \mathbf{1}_N\right\rangle. \end{aligned} \tag{21.64}$$

As with the strong constraint probabilistic approach, it is possible to apply this theory to the mixed distribution where we would have Gaussian and lognormally distributed model errors that we wish to minimize throughout the assimilation window. See [129] for the details about the mixed model errors and some results with the Lorenz 63 model.

21.8 Results of the Lorenz 1963 Model for 4D VAR

To illustrate the application of the mixed Gaussian-lognormal 4D VAR scheme, we present some results with the Lorenz 1963 model, where we compare the performance of the mixed distribution approach with the transform approach. We shall present results for the strong constraint formulations, but present the set up for both strong and weak constraint mixed distribution 4D VAR. These results are from [129].

$$J_{mx}\left(\mathbf{x}_0^t\right) = \frac{1}{2}\boldsymbol{\varepsilon}_0^{b^T}\ \mathbf{B}_0^{-1}\boldsymbol{\varepsilon}_0^b + \frac{1}{2}\sum_{i=1}^{N_o}\boldsymbol{\varepsilon}_i^{o^T}\ \mathbf{R}_i^{-1}\boldsymbol{\varepsilon}_i^o + \left(\ln z_0 - \ln z_0^b\right) + \sum_{i=1}^{N_o}\left(\ln z_i^o - \ln \boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right)\right), \tag{21.65}$$

$$J_{tr}\left(\mathbf{X}_0\right) = \frac{1}{2}\hat{\boldsymbol{\varepsilon}}_0^{b^T}\ \mathbf{B}_0^{-1}\hat{\boldsymbol{\varepsilon}}_0^b + \frac{1}{2}\sum_{i=1}^{N_o}\hat{\boldsymbol{\varepsilon}}_i^{o^T}\ \mathbf{R}_i^{-1}\hat{\boldsymbol{\varepsilon}}_i^o, \tag{21.66}$$

where

$$\boldsymbol{\varepsilon}_0^b = \begin{pmatrix} x_0 - x_0^b \\ y_0 - y_0^b \\ \ln z_0 - \ln z_0^b \end{pmatrix},\ \boldsymbol{\varepsilon}_i^o = \begin{pmatrix} x_i^o - \boldsymbol{\mathcal{M}}_{0,i}\left(x_0\right) \\ y_i^o - \boldsymbol{\mathcal{M}}_{0,i}\left(y_0\right) \\ \ln z_i^0 - \ln \boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right) \end{pmatrix},$$

$$\hat{\boldsymbol{\varepsilon}}_0^b = \begin{pmatrix} x_0 - x_{b,0} \\ y_0 - y_{b,0} \\ Z_0 - Z_{b,0} \end{pmatrix},\quad \hat{\boldsymbol{\varepsilon}}_i^o = \begin{pmatrix} x_i^o - \boldsymbol{\mathcal{M}}_{0,i}\left(x_0\right) \\ y_i^o - \boldsymbol{\mathcal{M}}_{0,i}\left(y_0\right) \\ Z_i^0 - Z_i \end{pmatrix},$$

$$\mathbf{x}^T = \begin{pmatrix} x & y & z \end{pmatrix},\qquad \mathbf{X}^T = \begin{pmatrix} x & y & \ln z \end{pmatrix},$$

where $Z = \ln z$.

The gradients of (21.65) and (21.66) are

$$\nabla_{\mathbf{x}_0^t} J_{hy} = \mathbf{W}_b^T\left(\mathbf{B}_0^{-1}\boldsymbol{\varepsilon}_0^b - \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\right) + \sum_{i=1}^{N_o}\mathbf{M}_{0,i}^T\left(\mathbf{W}_{o,i}^T\left(\mathbf{R}_i^{-1}\boldsymbol{\varepsilon} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}\right)\right), \tag{21.67}$$

$$\nabla_{\mathbf{X}_0^t} J_{tr} = \mathbf{B}_0^{-1}\hat{\boldsymbol{\varepsilon}}_0^b + \mathbf{W}_b^{-T}\sum_{i=1}^{N_o}\mathbf{M}_i^T\ \mathbf{R}_i^{-1}\hat{\boldsymbol{\varepsilon}}_i^o, \tag{21.68}$$

where

$$\mathbf{W}_b = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & z_0^{-1} \end{pmatrix},\quad \mathbf{W}_{o,i} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \left(\boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right)\right)^{-1} \end{pmatrix},$$

and we can see the offset term in (21.67) for the mode for the lognormal component.

Another important feature to note about (21.67) and (21.68) is that the increment the tangent linear model is applied to is different for the two approaches t. For (21.67) we have $\delta\mathbf{x}$ equal to

$$\delta\mathbf{x}_i = \begin{pmatrix} \sigma_{o,x}^{-2}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(x_0\right) - x_i^o\right) \\ \sigma_{o,y}^{-2}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(y_0\right) - y_i^o\right) \\ \left(\sigma_{o,z}^{-2}\left(\ln \boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right) - \ln z_i^o\right) + 1\right)\left(\boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right)\right)^{-1} \end{pmatrix}, \tag{21.69}$$

while for (21.68) $\delta\mathbf{x}$ is equal to

$$\delta\mathbf{x}_i = \begin{pmatrix} \sigma_{o,x}^{-2}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(x_0\right) - x_i^o\right) \\ \sigma_{o,y}^{-2}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(y_0\right) - y_i^o\right) \\ \sigma_{o,z}^{-2}\left(\ln \boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right) - \ln z_i^o\right)\left(\boldsymbol{\mathcal{M}}_{0,i}\left(z_0\right)\right)^{-1} \end{pmatrix}. \tag{21.70}$$

We can see that the increments are quite different when the observational error variance is large, but are similar for smaller values. This feature is consistent with the difference between the mode and the median of a lognormal distribution, but it shows that the mixed distribution approach is fitting the mode throughout the assimilation windows to reduce the innovations while the transform approach is fitting the best median to reduce the innovations.

Now we arrive at an important feature of lognormal distribution-based variational data assimilation: the defining of the background error covariance matrix. When the original work was being undertaken for the development and testing of the lognormal, and mixed distribution approach, it became apparent that using a climatological averaged background error covariance matrix was not aiding the assimilation scheme. It became quite clear that the **B** matrix had to be **updated** after each assimilation window or cycle.

To minimize the cost functions, the MATLAB® routine FMINSEARCH, which uses a Nelder-Mead simplex direct search method to find the minimum, is used. The routine is given the cost function and the gradient to evaluate.

The background error covariance matrices here are evolved through the assimilation experiment. The generation of the matrices are as follows: the initial covariance matrix is calculated as the average of the differences between the true solution of a background solution (whose initial conditions are $x_{b,0} = -5.9$, $y_{b,0} = -5.0$, and $z_{b,0} = 24.0$, whereas the true solutions starts from $x_{t,0} = -5.4458$, $y_{t,0} = -5.4841$, and $z_{t,0} = 22.5606$) over the length of the first assimilation window. As the matrix is only a 3×3 matrix, the exact inverse matrix is used. The true and the background trajectories can be seen in Fig. 21.8.

The $\mathbf{B}_0$ matrices for the remaining assimilation windows are calculated as follows:

$$\begin{aligned} \mathbf{B}_0^{(1)} &= \frac{1}{S}\sum_{i=1}^{S}\left\langle \mathbf{x}^b - \mathbf{x}_{mx}, \left(\mathbf{x}^b - \mathbf{x}_{mx}\right)^T \right\rangle_i, \\ \mathbf{B}_0^{(2)} &= \frac{1}{S}\sum_{i=1}^{S}\left\langle \mathbf{x}_{mx}^{(1)} - \mathbf{x}_{mx}^{(2)}, \left(\mathbf{x}_{mx}^{(1)} - {}^{(2)}_{mx}\right)^T \right\rangle_i, \\ &\vdots \\ \mathbf{B}_0^{(k)} &= \frac{1}{S}\sum_{i=1}^{S}\left\langle \mathbf{x}_{mx}^{(k-1)} - \mathbf{x}_{mx}^{(k)}, \left(\mathbf{x}_{mx}^{(k-1)} - \mathbf{x}_{mx}^{(k)}\right)^T \right\rangle_i, \end{aligned} \tag{21.71}$$

where S is the total number of time steps in the assimilation window, $k = 2, 3, \ldots, K$, K is the total number of assimilation windows, and $\mathbf{x}_{mx}$ is the solution from forecast from the previous analysis time through the next assimilation window. Therefore the **B** matrices are calculated as the differences between the model run starting with the solution at the end of the last assimilation window and the

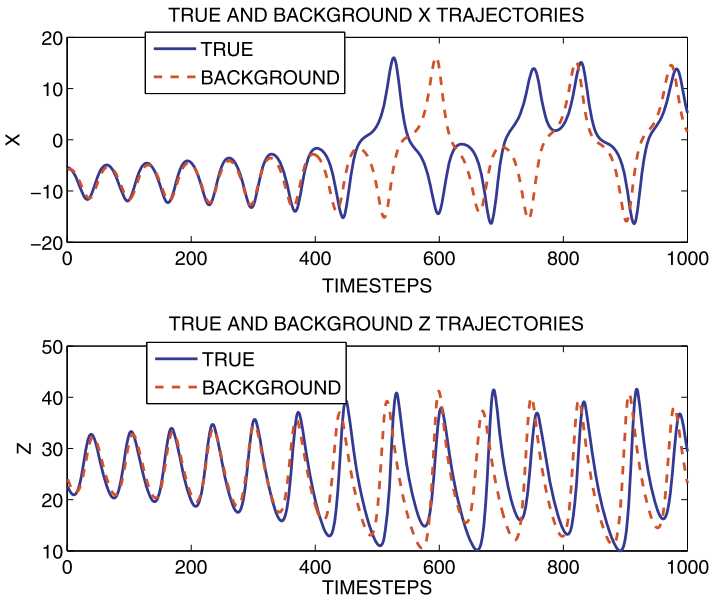


FIGURE 21.8

Plot of the true and the background trajectories for the Lorenz 1963 model experiments.

solution coming from the minimization for the current assimilation window. The same method is used to generate the **B** matrix for the transform approach.

In Fig. 21.9 we present results from [129] for the strong constraint form of 4D VAR for the experiments described above. We again see that when the observational error is quite small, there is not much difference in the performance of the transform and the mixed distribution approach; however, when the observational error is increased, then the mixed approach stays stable and consistent with the true trajectory, but the transform approach does not stay stable.

In [129] the authors also present results from using the full field mixed distribution for a weak constraint experiments with the Lorenz 1963 model. We now present the derivation of the equations that need to be solved for the state augmentation approach for a constant additive bias for the Gaussian components and a multiplicative bias for the lognormal component.

The version of weak constraint 4D VAR considered in [129] is the state augmentation approach as outlined in [503]. As a reminder, this approach is where the model error term is considered as a control variable, but where it is assumed that the errors associated with the initial conditions are independent of the model error terms. The assumption of the independence of the initial condition, and model, errors enables the $\mathbf{B}_0$ and $\mathbf{Q}$ matrices to be only size N_s, rather than $2N_s$. It is assumed that the model error term is constant for each time step but is initially randomly generated from a lognormal or Gaussian distribution initially. The cost function for the mixed Gaussian-lognormal weak constraint modal approach is given by

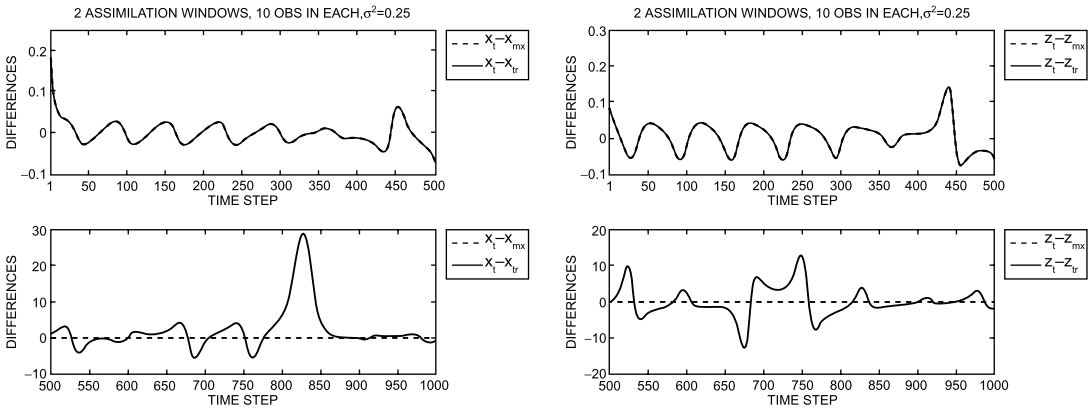


FIGURE 21.9

Results from experiments comparing the mixed distribution approach with the logarithmic approach using the Lorenz 1963 model for 4D full field VAR, Figure 9 from S.J. Fletcher (2010) Mixed Gaussian-lognormal four-dimensional data assimilation, Tellus A: Dynamic Meteorology and Oceanography, 62:3, 266-287, DOI: 10.1111/j.1600-0870.2009.00439.x. https://creativecommons.org/licenses/by/4.0/.

$$
\begin{aligned}
J\left(\mathbf{x}_0^t, \mathbf{\Phi}\right) = & \frac{1}{2}\begin{pmatrix} \mathbf{x}_{0,p} - \mathbf{x}_{b,0,p} \\ \ln \mathbf{x}_{0,q} - \ln \mathbf{x}_{b,0,q} \end{pmatrix}^T \mathbf{B}_0^{-1} \begin{pmatrix} \mathbf{x}_{0,p} - \mathbf{x}_{b,0,p} \\ \ln \mathbf{x}_{0,q} - \ln \mathbf{x}_{b,0,q} \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \mathbf{x}_{0,p} - \mathbf{x}_{b,0,p} \\ \ln \mathbf{x}_{0,q} - \ln \mathbf{x}_{b,0,q} \end{pmatrix}, \begin{pmatrix} \mathbf{0}_p \\ \mathbf{1}_q \end{pmatrix} \right\rangle \\
& + \frac{1}{2}\sum_{i=1}^{N_o} \begin{pmatrix} \mathbf{y}_{p_i,i} - \mathbf{y}_{p_i,i,k} \\ \ln \mathbf{y}_{q_i,i} - \ln \mathbf{y}_{q_i,i,k} \end{pmatrix}^T \mathbf{R}_i^{-1} \begin{pmatrix} \mathbf{y}_{p_i,i} - \mathbf{y}_{p_i,i,k} \\ \ln \mathbf{y}_{q_i,i} - \ln \mathbf{y}_{q_i,i,k} \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \mathbf{y}_{p_i,i} - \mathbf{y}_{p_i,i,k} \\ \ln \mathbf{y}_{q_i,i} - \ln \mathbf{y}_{q_i,i,k} \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p_i,i} \\ \mathbf{1}_{q_i,i} \end{pmatrix} \right\rangle \\
& + \frac{1}{2}\begin{pmatrix} \mathbf{\Phi}_p \\ \ln \mathbf{\Phi}_q \end{pmatrix}^T \mathbf{Q}^{-1} \begin{pmatrix} \mathbf{\Phi}_p \\ \ln \mathbf{\Phi}_q \end{pmatrix} + \left\langle \begin{pmatrix} \mathbf{\Phi}_p \\ \ln \mathbf{\Phi}_q \end{pmatrix}, \begin{pmatrix} \mathbf{0}_p \\ \mathbf{1}_q \end{pmatrix} \right\rangle,
\end{aligned} \tag{21.72}
$$

subject to

$$
\mathbf{x}_{k,p} = \boldsymbol{\mathcal{M}}_k\left(\mathbf{x}_{k-1}\right) + \mathbf{\Phi}_p, \tag{21.73a}
$$

$$
\left[\mathbf{x}_{k,q}\right]_s = \left[\boldsymbol{\mathcal{M}}_k\left(\mathbf{x}_{k-1}\right)\right]_s \left[\mathbf{\Phi}_q\right]_s, \quad s = p+1, p+2, \ldots, q, \tag{21.73b}
$$

$$
\mathbf{y}_{i,k} = \mathbf{h}_i\left(\mathbf{x}_k\right). \tag{21.73c}
$$

In Section 16.5 we showed how to obtain the gradient of a Gaussian-based state augmentation approach's cost function through using calculus of variation techniques. These same techniques can be used for the lognormal-based system. The full derivation of the variational approach for lognormal model errors can be found in Appendix B of [129], while the mixed distribution variational approach can be found in Appendix C in [129]. Here we present the Jacobian of (21.72), which can be verified to be

$$\nabla J = \begin{pmatrix} \nabla_{\mathbf{x}_0} J \\ \nabla_{\boldsymbol{\Phi}} J \end{pmatrix}, \tag{21.74}$$

$$\begin{aligned}\nabla_{\mathbf{x}_0^t} J = {} & \widehat{\mathbf{W}}_b^T \left(\mathbf{B}_0^{-1} \begin{pmatrix} \mathbf{x}_{0,p} - \mathbf{x}_{b,0,p} \\ \ln \mathbf{x}_{0,q} - \ln \mathbf{x}_{b,0,q} \end{pmatrix} + \begin{pmatrix} \mathbf{0}_p \\ \mathbf{1}_q \end{pmatrix} \right) \\ & - \sum_{i=1}^{N_o} \widetilde{\boldsymbol{\mathcal{M}}}_{1,k}^T \mathbf{H}_i^T \widehat{\mathbf{W}}_{o,i}^T \left(\mathbf{R}_i^{-1} \begin{pmatrix} \mathbf{y}_{p_i,i} - \mathbf{y}_{p_i,i,k} \\ \ln \mathbf{y}_{q_i,i} - \ln \mathbf{y}_{q_i,i,k} \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{p_i,i} \\ \mathbf{1}_{q_i,i} \end{pmatrix} \right)\end{aligned} \tag{21.75}$$

$$\begin{aligned}\nabla_{\boldsymbol{\Phi}} J = {} & - \sum_{i=1}^{N_0} \left(\widetilde{\boldsymbol{\mathcal{M}}}_k^T + \sum_{n=0}^{k-2} \widetilde{\boldsymbol{\mathcal{M}}}_{k-n-1}^T \widetilde{\mathbf{M}}_{k-n,k}^T \right) \mathbf{H}_i^T \widehat{\mathbf{W}}_{o,i}^T \left(\mathbf{R}_i^{-1} \begin{pmatrix} \mathbf{y}_{p_i,i} - \mathbf{y}_{p_i,i,k} \\ \ln \mathbf{y}_{q_i,i} - \ln \mathbf{y}_{q_i,i,k} \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{p_i,i} \\ \mathbf{1}_{q_i,i} \end{pmatrix} \right) \\ & + \widehat{\mathbf{W}}_m \left(\mathbf{Q}^{-1} \begin{pmatrix} \boldsymbol{\Phi}_p \\ \ln \boldsymbol{\Phi}_q \end{pmatrix} + \begin{pmatrix} \mathbf{0}_p \\ \mathbf{1}_q \end{pmatrix} \right),\end{aligned} \tag{21.76}$$

where

$$\widetilde{\mathbf{M}}_k = \begin{pmatrix} \mathbf{M}_{k,p,p} & \mathbf{M}_{k,p,q} \\ \widehat{\mathbf{M}}_{k,q,p} & \widehat{\mathbf{M}}_{k,q,q} \end{pmatrix}, \quad \widetilde{\boldsymbol{\mathcal{M}}}_k = \begin{pmatrix} \mathbf{I}_{p,p} & \mathbf{0}_{p,q} \\ \mathbf{0}_{q,p} & \widehat{\boldsymbol{\mathcal{M}}}_{k,q,q} \end{pmatrix},$$

$$\left[\widehat{\boldsymbol{\mathcal{M}}}_k\right]_{i,j} = \frac{\partial \boldsymbol{\mathcal{M}}_k \left(\mathbf{x}_{k-1}\right)_i}{\partial x_j} \Phi_i, \quad i = 1, 2, \ldots, N_s, \ j = 1, 2, \ldots, N_s$$

$$\left[\widehat{\boldsymbol{\mathcal{M}}}_k\right]_{i,j} = \boldsymbol{\mathcal{M}}_k \left(\mathbf{x}_{k-1}\right)_i \frac{\partial \Phi_i}{\partial \Phi_j}, \quad i = 1, 2, \ldots, N_s, \ j = 1, 2 \ldots, N_s,$$

and

$$\hat{\mathbf{W}}_m = \begin{pmatrix} \mathbf{I}_p & \mathbf{0}_{p,q} \\ \mathbf{0}_{q,p} & \operatorname{diag}\left(\boldsymbol{\Phi}_j^{-1}\right) \end{pmatrix}, \quad j = p+1, p+2, \ldots, p+q.$$

As we can see from the expression above, when we try to minimize the model error component with a lognormal distribution we have to take into account that in the differentiation we have a product of the numerical model and the model error. The effect that this has on the variational approach is shown in [129]. There the authors show that as the window length increases, the transform approach is not as capable in reducing the effect that the model error has on the analysis.

All of the theory presented so far in this chapter for the mixed Gaussian-lognormal distribution has been for full field 3D and 4D VAR, which is applicable in smaller dimensional problems; however, as we showed in Section 16.3, the operational numerical weather, and some ocean, prediction centers use the incremental formulation. As such, we now move on to non-Gaussian incremental 3D and 4D VAR.

21.9 Incremental Lognormal and Mixed 3D and 4D VAR

The first problem that arises when trying to formulate the incremental version of 3D and 4D VAR is the property that the difference between two lognormal random variables is not a lognormal random variable. Therefore, given that the lognormal distribution is a geometric distribution, we need to form an increment that maintains the ratio property of lognormal random variables.

The first attempt to incrementalize the mixed lognormal-Gaussian 4D VAR cost function from [129] appears in [409]. The basis of the approach in [409] is to define an increment, $\Delta\mathbf{x}$, as

$$\mathbf{x}_0^t \equiv \mathbf{x}_0^b \circ \exp\{\Delta\mathbf{x}\}, \tag{21.77}$$

where $\mathbf{x}_0^t$ is the true state, $\mathbf{x}_b$ is the background state, and $\Delta\mathbf{x}$ is the *increment* that is sought and $\circ$ represents the Hadamard componentwise multiplication operator as before.

A condition that arises that enables both sides in (21.77) to be equivalent lognormal random variables is the product property mentioned in the introduction, which is

$$\underbrace{\mathbf{x}_0^t}_{\sim LN} = \underbrace{\mathbf{x}^b}_{\sim LN} \circ \underbrace{\exp\{\Delta\mathbf{x}\}}_{\sim LN}. \tag{21.78}$$

However, for the second term on the right-hand side of (21.78) to be lognormal then another property applies here; that is if $\mathbf{x} \sim G(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ then $\exp\{\mathbf{x}\} \sim LN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Therefore, the only way that (21.78) can be valid is if $\Delta\mathbf{x}$ is Gaussian distributed.

The next step in the incremental derivation from [409] is to substitute the increment in (21.77) into the full field mixed distribution modal strong constraint 4D VAR cost function (21.57). However, as the authors show in [409], this approach is not as optimal as using the transform cost function. As shown earlier in this chapter, the median approach involves solving the Gaussian cost function, but as we have just stated, the increment defined in (21.78) is a Gaussian increment. Therefore, the median approach in [409] is the correct one to solve with (21.78).

This raises the question about the point of the modal approach that has been developed throughout the Fletcher papers; why did modal approach not beat the median/transform approach with the increment from (21.78)? The answer is because the increment in (21.78) must be a Gaussian random variable for (21.77) to be defining a lognormal random variable. Therefore, the incremental approach presented in [409] is equivalent to a log-linearization of the lognormal cost function. However, this now raises a difference between the Gaussian incremental approach and how we would form a lognormal modal-based incremental formulation.

The work in [409] highlighted the fact that we cannot simply linearize the lognormal modal cost function with (21.78). It implies that the increment is not consistent with the underlying Bayesian model. However, the transform approach is and that is why the results were better with that cost function. This means that when considering non-symmetric distributions for future non-Gaussian incremental VAR formulations, **we need to form the Bayesian model for the increment, not linearize the Bayesian model**.

In [132] an alternative incremental formulation for a lognormal distribution is defined, such it is possible to perform linearizations consistent with the geometric behavior of the lognormal distribution. The starting point for deriving multiplicative incremental VAR is to focus on the background errors. The original definition for the lognormal background errors is

$$\mathbf{x}_t = \mathbf{x}_b \circ \boldsymbol{\varepsilon}_L, \tag{21.79}$$

where $\boldsymbol{\varepsilon}_L$ is the lognormal background error. In the Gaussian framework the background error, $\boldsymbol{\varepsilon}_G$, is

$$\mathbf{x}_t = \mathbf{x}_b + \boldsymbol{\varepsilon}_G. \tag{21.80}$$

The basis of the Gaussian incremental VAR formulation is to rearrange (21.80) such that $\boldsymbol{\varepsilon}_G = \delta \mathbf{x}_G \equiv \mathbf{x}_t - \mathbf{x}_b$, where $\delta \mathbf{x}_G$ is the increment that is sought. For multiplicative incremental VAR, $\boldsymbol{\varepsilon}_L$ is defined as

$$\boldsymbol{\varepsilon}_L = \Delta \mathbf{x}_L \equiv \frac{\mathbf{x}_t}{\mathbf{x}_b} \Rightarrow \mathbf{x}_t = \mathbf{x}_b \circ \Delta \mathbf{x}_L. \tag{21.81}$$

An important feature of (21.81) is that the increment, $\Delta \mathbf{x}_L$, is lognormally distributed through the property that the ratio of two independent lognormal random variables is also a lognormal random variable, indicating a major difference from the approach in [409]. Therefore, we now require a Bayesian problem for this lognormally distributed random variable. For the remainder of this chapter we shall drop the subscripts on δ and Δ, but note that δ represents an additive increment and Δ represents a multiplicative increment.

21.9.1 Multiplicative Incremental 3D VAR

The first step in the derivation of the lognormal incremental VAR cost function is to substitute (21.81) into the background component of the lognormal full field cost function, which results in

$$J_B(\mathbf{x}) = \frac{1}{2} (\ln \mathbf{x}_t - \ln \mathbf{x}_b)^T \mathbf{B}_L^{-1} (\ln \mathbf{x}_t - \ln \mathbf{x}_b) + (\ln \mathbf{x}_t - \ln \mathbf{x}_b)^T \mathbf{1}_N, \tag{21.82}$$

$$= \frac{1}{2} (\ln \Delta \mathbf{x})^T \mathbf{B}_L^{-1} (\ln \Delta \mathbf{x}) + (\ln \Delta \mathbf{x})^T \mathbf{1}_N, \tag{21.83}$$

where $\mathbf{B}_L$ is the lognormal background error covariance matrix, $\mathbf{1}_N$ is a vector of 1s of length N, and N is the total number of background elements. Therefore, (21.83) is a consistent lognormal minimization problem in $\Delta \mathbf{x}$.

While the incrementalization of the lognormal background component is quite simple to achieve, the incrementalization of the observational component on the other hand is not so straightforward. As we saw for the Gaussian incremental formalization, we approximate the observation operator at the true state through what appears to be a Taylor series expansion for the nonlinear observation operator about the background state and then we have the Jacobian term. The expression used for the expansion of the observation operator is in fact a tangent linear approximation.

This means that for a lognormal multiplicative incremental approach we require the geometric tangent linear model introduced in Section 13.2 to linearize the logarithm of the observation operator for 3D VAR, and the logarithm of the observation operator operating on the nonlinear numerical model for 4D VAR. Therefore, applying the theorems from Section 13.2, results in the incremental lognormal-based 3D VAR cost function as

$$\begin{aligned} J(\Delta \mathbf{x}) = & \frac{1}{2} (\ln \mathbf{x}_t - \ln \mathbf{x}_b)^T \mathbf{B}_L^{-1} (\ln \mathbf{x}_t - \ln \mathbf{x}_b) + (\ln \mathbf{x}_t - \ln \mathbf{x}_b)^T \mathbf{1}_N \\ & + \frac{1}{2} (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}_t))^T \mathbf{R}_L^{-1} (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}_t)) + (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}_t))^T \mathbf{1}_{N_o}, \\ \equiv & \frac{1}{2} (\ln \Delta \mathbf{x})^T \mathbf{B}_L^{-1} (\ln \Delta \mathbf{x}) + (\ln \Delta \mathbf{x})^T \mathbf{1}_N \\ & + \frac{1}{2} \left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}_b) - \mathbf{W}_o^{-1} \mathbf{H} \left[\mathbf{x}_b \circ (\Delta \mathbf{x} - \mathbf{1}_{N_o}) \right] \right)^T \\ & \times \mathbf{R}_L^{-1} \left(\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}_b) - \mathbf{W}_o^{-1} \mathbf{H} \left[\mathbf{x}_b \circ (\Delta \mathbf{x} - \mathbf{1}_{N_o}) \right] \right) \end{aligned} \tag{21.84}$$

$$+\left(\ln \mathbf{y}-\ln \mathbf{h}\left(\mathbf{x}_b\right)-\mathbf{W}_o^{-1}\mathbf{H}\left[\mathbf{x}_b\circ\left(\Delta\mathbf{x}-\mathbf{1}_{N_o}\right)\right]\right)^T\mathbf{1}_{N_o}. \tag{21.85}$$

An important feature of (21.85) is that it is consistent with a modal approach for the lognormal increment, $\Delta\mathbf{x}$, which is different from the approach in [409]; that is to say that the solution to the Jacobian of (21.85) equal to zero is consistent with a mode of a multivariate lognormal posterior distribution. The justification of this statement is as a result of

$$\ln \mathbf{h}\left(\mathbf{x}_t\right)\equiv \ln \mathbf{h}\left(\mathbf{x}_b\circ\Delta\mathbf{x}\right)\approx \ln\left(\mathbf{h}\left(\mathbf{x}_b\right)\circ\exp\left\{\mathbf{W}_o^{-1}\mathbf{H}\left[\mathbf{x}_b\circ\left(\Delta\mathbf{x}-\mathbf{1}_{N_o}\right)\right]\right\}\right), \tag{21.86}$$

$$\approx \ln \mathbf{h}\left(\mathbf{x}_b\right)+\mathbf{W}_o^{-1}\mathbf{H}\left[\mathbf{x}_b\circ\left(\Delta\mathbf{x}-\mathbf{1}_{N_o}\right)\right], \tag{21.87}$$

and using the product property of the lognormal distribution, implies that the linearized problem is approximately lognormal.

The Jacobian of (21.85) is

$$\begin{aligned}\nabla J\left(\Delta\mathbf{x}\right)=&\,\Delta\mathbf{W}_b^{-T}\left(\mathbf{B}_L^{-1}\left(\ln\Delta\mathbf{x}\right)+\mathbf{1}_N\right)\\&-\mathbf{W}_B^T\mathbf{H}^T\mathbf{W}_o^{-T}\mathbf{R}_L^{-1}\left(\ln\mathbf{y}-\ln\mathbf{h}\left(\mathbf{x}_b\right)-\mathbf{W}_o^{-1}\mathbf{H}\left[\mathbf{x}_b\circ\left(\Delta\mathbf{x}-\mathbf{1}_{N_o}\right)\right]+\mathbf{1}_{N_o}\right),\end{aligned} \tag{21.88}$$

where $\Delta\mathbf{W}_b=\operatorname{diag}\left\{\Delta\mathbf{x}_i\right\}$ and $\mathbf{W}_b=\operatorname{diag}\left\{\mathbf{x}_{b,i}\right\}$ for $i=1,2,\ldots,N$.

21.9.2 Multiplicative Incremental 4D VAR

As we have seen for the Gaussian case and for the lognormal/mixed distribution case, the main difference between 3D VAR and 4D VAR is the inclusion of the temporal dimension. In Section 21.6 we showed that the 4D VAR cost function is equivalent to a maximum likelihood solution of a multievent Bayesian problem which incorporates the time component. Therefore, as mentioned at the start of this section, we are seeking a cost function that is consistent with the Bayesian problem for finding an increment that is consistent with a lognormal distribution's mode.

The starting point for deriving a lognormal incremental 4D VAR is the linearization, with respect to the multiplicative increment, of $\ln h_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(x_{t,0}\right)\right)$, where $\boldsymbol{\mathcal{M}}_{0,i}$ is the nonlinear numerical model between time $t=t_0$ and $t=t_i$. The linearization for this function is proven in [132], which yields

$$\ln h_i\left(\boldsymbol{\mathcal{M}}\left(x_{b,0}\Delta x\right)\right)\approx \ln h_i\left(\boldsymbol{\mathcal{M}}\left(x_{b,0}\right)\right)+\left(h_i\left(\boldsymbol{\mathcal{M}}_{0,i}\left(x_{t,0}\right)\right)\right)^{-1}\mathbf{H}_i\mathbf{M}_{0,i}\left[x_{b,0}\left(\Delta x-1\right)\right]. \tag{21.89}$$

Therefore, taking (21.89) and extending this to the multivariate case yields the lognormal-based multiplicative incremental 4D VAR cost function as

$$\begin{aligned}J\left(\Delta x\right)=&\,\frac{1}{2}\left(\ln\Delta\mathbf{x}_0\right)^T\mathbf{B}_L^{-1}\left(\ln\Delta\mathbf{x}_0\right)+\left(\ln\Delta\mathbf{x}_0\right)^T\mathbf{1}_N\\&+\frac{1}{2}\sum_{i=1}^{N_{o,T}}\left(\ln\mathbf{y}_i-\ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{o,i}\left(\mathbf{x}_{b,0}\right)\right)-\mathbf{W}_{o,i}^{-1}\mathbf{H}_i\mathbf{M}_{0,i}\left[\mathbf{x}_{b,0}\circ\left(\Delta\mathbf{x}_0-1\right)\right]\right)^T\mathbf{R}_{L,i}^{-1}\\&\times\left(\ln\mathbf{y}_i-\ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{o,i}\left(\mathbf{x}_{b,0}\right)\right)-\mathbf{W}_{o,i}^{-1}\mathbf{H}_i\mathbf{M}_{0,i}\left[\mathbf{x}_{b,0}\circ\left(\Delta\mathbf{x}_0-1\right)\right]\right)\\&+\sum_{i=1}^{N_{o,T}}\left(\ln\mathbf{y}_i-\ln\mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{o,i}\left(\mathbf{x}_{b,0}\right)\right)-\mathbf{W}_{o,i}^{-1}\mathbf{H}_i\mathbf{M}_{0,i}\left[\mathbf{x}_{b,0}\circ\left(\Delta\mathbf{x}_0-1\right)\right]\right)^T\mathbf{1}_{N_o,i},\end{aligned} \tag{21.90}$$

which we can see is fitting a lognormal modal increment to the observations in time.

The Jacobian of (21.90) can easily be shown to be

$$\begin{aligned}\nabla J\left(\Delta \mathbf{x}_0\right) = & \Delta \mathbf{W}_b^T\left(\mathbf{B}_L^{-1}\left(\ln \Delta \mathbf{x}_0\right) + \mathbf{1}_N\right) \\ & - \sum_{i=1}^{N_{o,T}} \mathbf{W}_b^T\left(\mathbf{M}_{o,i}^T \mathbf{H}_i^T \mathbf{W}_{o,i}^{-T}\left(\mathbf{R}_{L,i}^{-1}\left(\ln \mathbf{y}_i - \ln \mathbf{h}_i\left(\boldsymbol{\mathcal{M}}_{o,i}\left(\mathbf{x}_{b,0}\right)\right)\right.\right.\right. \\ & \left.\left.\left. - \mathbf{W}_{o,i}^{-1} \mathbf{H}_i \mathbf{M}_{0,i}\left[\mathbf{x}_{b,0} \circ \left(\Delta \mathbf{x}_0 - 1\right)\right]\right) + \mathbf{1}_{N_{o,i}}\right)\right).\end{aligned} \tag{21.91}$$

As with the Jacobian of the lognormal incremental 3D VAR cost function, we can clearly see the lognormal modal structure for both the background and the observational components in (21.91).

21.9.3 Mixed Additive and Multiplicative Incremental VAR

As in [129,136,137], and consistent with how we have progressed through the summary of this work in this chapter, the next step is to combine the associated lognormal theory with Gaussian random variable theory. While for the full field versions of VAR data assimilation the starting point is to define the errors, here we have to define increments that follow the mixed distribution from [136]. Thus, these increments are define by

$$\mathbf{x}^t = \begin{pmatrix} \mathbf{x}_p^t & \sim G \\ \mathbf{x}_q^t & \sim LN \end{pmatrix} \equiv \begin{pmatrix} \mathbf{x}_{b,p} + \delta \mathbf{x}_p \\ \mathbf{x}_{b,q} \circ \Delta \mathbf{x}_q \end{pmatrix} \Rightarrow \begin{pmatrix} \mathbf{x}_p^t - \mathbf{x}_{b,p} \\ \frac{\mathbf{x}_q^t}{\mathbf{x}_{b,q}} \end{pmatrix} = \begin{pmatrix} \delta \mathbf{x}_p \\ \Delta \mathbf{x}_q \end{pmatrix} \equiv \Delta \mathbf{x}_{mx}, \tag{21.92}$$

where p is the number of Gaussian additive increments, and q is the number of lognormal multiplicative increments. For us to be able to linearize the observational component for the 3D VAR derivation, we require properties of both the additive and multiplicative tangent linear models. Therefore, the mixed distribution incremental 3D VAR cost function can be shown to be

$$\begin{aligned} J\left(\Delta \mathbf{x}_{mx}\right) = & \frac{1}{2}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \ln \Delta \mathbf{x}_{b,q} \end{pmatrix}^T \mathbf{B}_{mx}^{-1}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \ln \Delta \mathbf{x}_{b,q} \end{pmatrix} + \left\langle \begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \ln \Delta \mathbf{x}_{b,q} \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{b,p} \\ \mathbf{1}_{b,q} \end{pmatrix} \right\rangle \\ & + \frac{1}{2}\begin{pmatrix} \mathbf{y}_{o,p} - \mathbf{h}_{o,p}\left(\mathbf{x}_b\right) - \mathbf{H}_{o,p}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta \mathbf{x}_q - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q} - \ln \mathbf{h}_{o,q}\left(\mathbf{x}_b\right) - \mathbf{W}_{o,q}^{-1} \mathbf{H}_{o,q}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta \mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix}^T \mathbf{R}_{mx}^{-1} \\ & \times \begin{pmatrix} \mathbf{y}_{o,p} - \mathbf{h}_{o,p}\left(\mathbf{x}_b\right) - \mathbf{H}_{o,p}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta \mathbf{x}_q - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q} - \ln \mathbf{h}_{o,q}\left(\mathbf{x}_b\right) - \mathbf{W}_{o,q}^{-1} \mathbf{H}_{o,q}\begin{pmatrix} \delta \mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta \mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix} \end{aligned}$$

$$+\left\langle \begin{pmatrix} \mathbf{y}_{o,p} - \mathbf{h}_{o,p}\left(\mathbf{x}_b\right) - \mathbf{H}_{o,p}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_q - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q} - \ln \mathbf{h}_{o,q}\left(\mathbf{x}_b\right) - \mathbf{W}_{o,q}^{-1}\mathbf{H}_{o,q}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{o,p} \\ \mathbf{1}_{o,q} \end{pmatrix} \right\rangle. \tag{21.93}$$

Given the definition for the mixed additive and multiplicative increments in (21.92), it is quite simple to alter these to those for a 4D VAR system, and thus the mixed multiplicative-additive 4D VAR cost function can be shown to be

$$\begin{aligned} J\left(\Delta\mathbf{x}_{mx}\right) = {} & \frac{1}{2}\begin{pmatrix} \delta\mathbf{x}_{b,p}\left(t_0\right) \\ \ln \Delta\mathbf{x}_{b,q}\left(t_0\right) \end{pmatrix}^T \mathbf{B}_{mx}^{-1}\begin{pmatrix} \delta\mathbf{x}_{b,p}\left(t_0\right) \\ \ln \Delta\mathbf{x}_{b,q}\left(t_0\right) \end{pmatrix} + \begin{pmatrix} \delta\mathbf{x}_{b,p}\left(t_0\right) \\ \ln \Delta\mathbf{x}_{b,q}\left(t_0\right) \end{pmatrix}^T \begin{pmatrix} \mathbf{0}_{b,p} \\ \mathbf{1}_{b,q} \end{pmatrix} \\ & + \sum_{i=1}^{N_o} \frac{1}{2}\begin{pmatrix} \mathbf{y}_{o,p,i} - \mathbf{h}_{o,p,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{H}_{o,p,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q,i} - \ln \mathbf{h}_{o,q,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{W}_{o,q,i}^{-1}\mathbf{H}_{o,q,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix}^T \mathbf{R}_{mx,i}^{-1} \\ & \times \begin{pmatrix} \mathbf{y}_{o,p,i} - \mathbf{h}_{o,p,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{H}_{o,p,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q,i} - \ln \mathbf{h}_{o,q,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{W}_{o,q,i}^{-1}\mathbf{H}_{o,q,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix} \\ & + \sum_{i=1}^{N_o} \begin{pmatrix} \mathbf{y}_{o,p,i} - \mathbf{h}_{o,p,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{H}_{o,p,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \\ \ln \mathbf{y}_{o,q,i} - \ln \mathbf{h}_{o,q,i}\left(\boldsymbol{\mathcal{M}}_{0,i}\left(\mathbf{x}_{b,0}\right)\right) - \mathbf{W}_{o,q,i}^{-1}\mathbf{H}_{o,q,i}\mathbf{M}_{0,i}\begin{pmatrix} \delta\mathbf{x}_{b,p} \\ \mathbf{x}_{b,q} \circ \left(\Delta\mathbf{x}_{b,q} - \mathbf{1}_{b,q}\right) \end{pmatrix} \end{pmatrix}^T \begin{pmatrix} \mathbf{0}_{o,p} \\ \mathbf{1}_{o,q} \end{pmatrix}. \end{aligned} \tag{21.94}$$

21.9.4 Analysis Mean of a Lognormal Data Assimilation System Not Equal to Zero

In [132] the four-dimensional mixed cost function is tested with the Lorenz 1963 model, where experiments are performed to assess the robustness of certain features of the data assimilation scheme. It is shown that for small lognormal observational error variances, the analysis distribution looks approximately Gaussian, but as the observational error variance increases, the analysis distribution, which could be formed as the *true* solution exists for the experiments, becomes more lognormal in its appearance with a mode at 1. This may appear a bit confusing considering that it has been assumed that the background error and the observational error means for the lognormal component are zero. We have provided copies of some of the plots of this feature that appear in [132] in Fig. 21.10. We shall briefly explain why it is a sign that the lognormal-based data assimilation system is performing well if the analysis distribution for a lognormal random variable has a mode at 1 and as such the mean of the analysis distribution $\boldsymbol{\mu}_a$ cannot be zero.

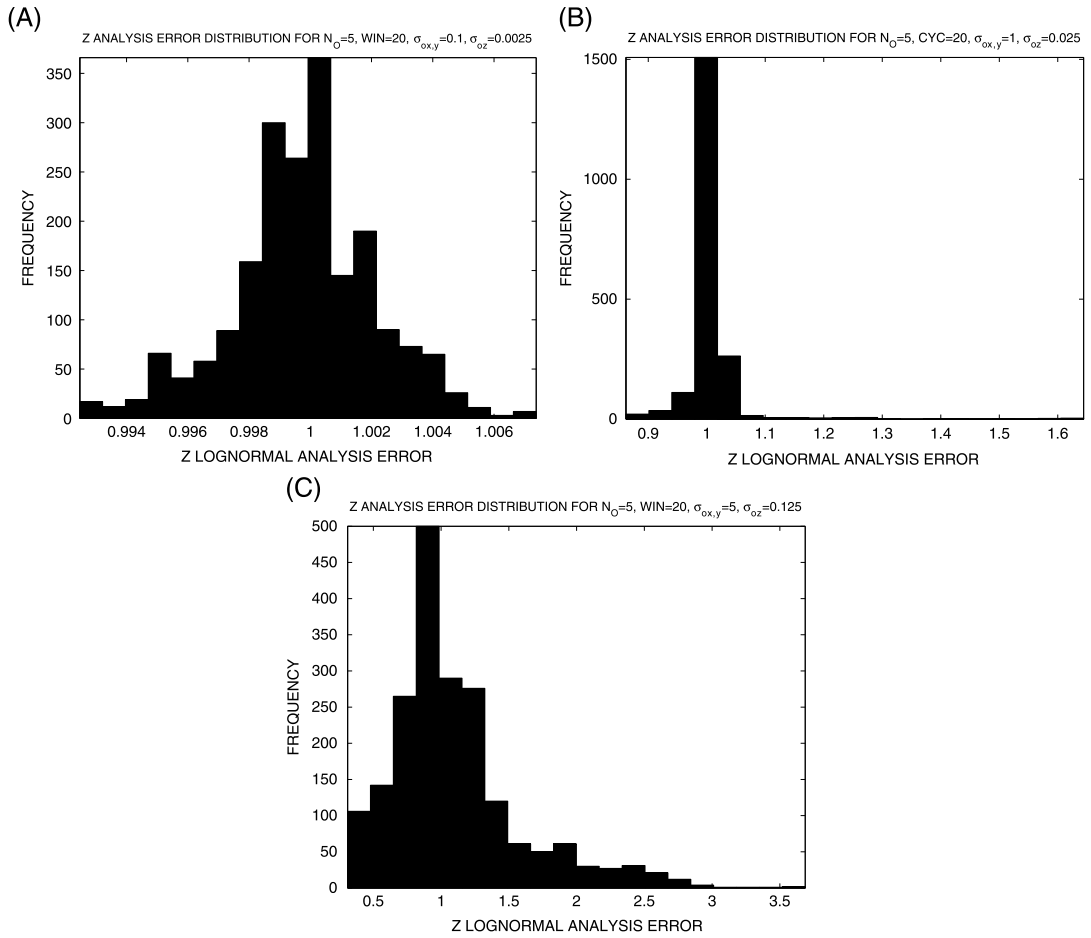


FIGURE 21.10

Plots of the analysis distribution of the z component for mixed incremental VAR for different configurations from Fletcher, S. J., and Jones, A. S. (2014). Multiplicative and Additive Incremental Variational Data Assimilation for Mixed Lognormal–Gaussian Errors, Monthly Weather Review, 142(7), 2521-2544. **©American Meteorological Society. Used with permission.**

The first step in understanding why the analysis mean cannot be zero is to recall that the analysis error is defined in terms of **ratios, not differences**. Thus, if the mode of the posterior distribution is approximately equal to the true solution, this would imply that the most likely value for the ratio of the two random variables is 1. However, to obtain a distribution with a mode of 1, the random variable that the distribution is representing cannot have a mean of zero, recalling that the lognormal distribution is defined by the expectation of $\ln x$, **not** x. We now prove this statement below.

We first use the fact that the posterior distribution is a lognormal, as the product of two lognormal distributions is itself a lognormal distribution, $x_{post} \sim LN\left(\mu_{post}, \sigma^2_{post}\right)$. This can easily be proven using the same technique for the Gaussian equivalent. Therefore, if we assume that the true state's distribution is also lognormal, $x_{true} \sim LN\left(\mu_{true}, \sigma^2_{true}\right)$, then the distribution of the ratio of two lognormal independent random variables is also a lognormal distribution, $\varepsilon_a \equiv x_{true}/x_{post} \sim LN\left(\mu_{true} - \mu_{post}, \sigma^2_{true} + \sigma^2_{post}\right)$ [80].

The next step is to consider the definition of the mode of a lognormal distribution

$$MLE\,(x) = \exp\left\{\mu - \sigma^2\right\}. \tag{21.95}$$

Substituting for the analysis error mean and variance into (21.95) yields the expression for the analysis error mode as

$$MLE\,(\varepsilon) \equiv \exp\left\{\mu_{true} - \mu_{post} - \sigma^2_{true} - \sigma^2_{post}\right\}. \tag{21.96}$$

For the expression in (21.96) to be equal to 1, we require the power of the exponential to be zero. Therefore, if the analysis error is *unbiased*, then $\mu_{true} - \mu_{post}$ should be equal to zero. However, if that were to occur, then the only way that the remaining terms could make the analysis error mode be 1 would be if $\sigma^2_{true} = \sigma^2_{post} = 0$, which cannot happen. Therefore, the analysis error mean cannot be zero, which then implies that the analysis distribution will be biased in Gaussian space if the data assimilation scheme is performing well.

To finish this explanation, we consider the fact that if there was a bias with the solution to the Bayesian problem, then it would show up in the form of an analysis error distribution that did not have a mode at 1. If the solution was always underestimating the true state, then the analysis error distribution's mode would be greater than 1. The analysis error distribution's mode would be less than 1 if the solution to the Bayesian problem was frequently overestimating the true state.

21.9.5 Comparison of a Mixed Incremental System With Gaussian-Only Scheme

One of the sets of experiments presented in [132] is a comparison of the mixed multiplicative-additive incremental approach with the Lorenz 63 model with a full Gaussian-based additive incremental formulation. One of the important findings from this experiment is the fact that when the lognormal observational variance is small, there was no sizable difference in the performance of the mixed approach to the Gaussian one; however, when the observational error variance grew, the mixed approach was able to stay on the correct attractor, while the Gaussian-based approach did not and was not able to return to the correct attractor throughout the rest of the experiment. An example of this behavior can be seen in Fig. 21.11 from [132].

This indicates that when the lognormal observational error variance is small, so that skewness is small, it is possible to approximate this situation with a Gaussian incremental VAR system; however, as this variance increases this is not the case, corresponding to the skewness increasing, and as such a mixed approach should be considered.

However, as we alluded to earlier, there could still be ways to optimize the lognormal-based data assimilation systems further by selecting the correct descriptive statistic to minimize the errors **at that analysis time**. While throughout this chapter we have advocated that the mode should be the basis of

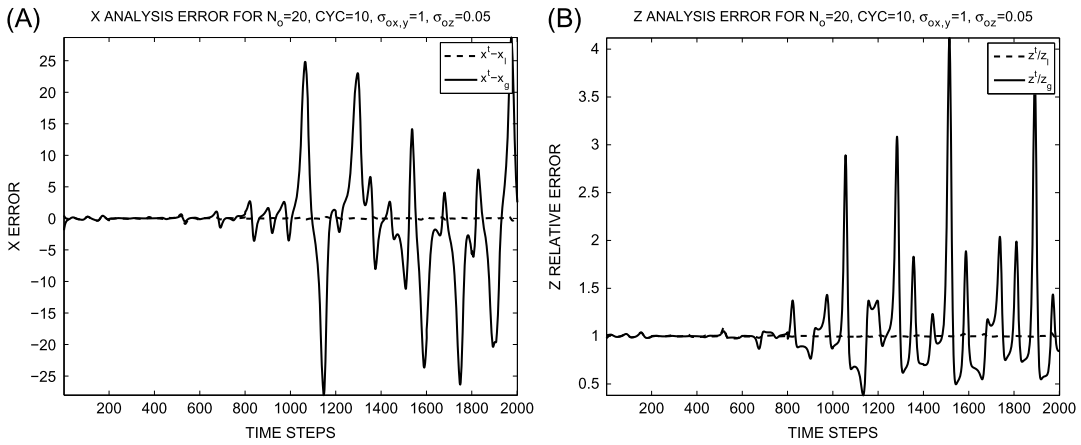


FIGURE 21.11

Figures from Fletcher and Jones (2014) comparing the mixed incremental approach against the Gaussian increment fits all approach. From Fletcher, S. J., and Jones, A. S. (2014). Multiplicative and Additive Incremental Variational Data Assimilation for Mixed Lognormal–Gaussian Errors, Monthly Weather Review, 142(7), 2521-2544. **© American Meteorological Society. Used with permission.**

the lognormal, and hence the mixed Gaussian-lognormal VAR system; however, when implementing the full field mixed 1D VAR approach into a temperature-mixing ratio microwave retrieval system [221] provided some interesting features that had not been expected. Part of this surprise occurred due to the mindset that all of the properties of the Gaussian transfer to other distributions, but this is not the case. We shall present a summary of these findings in the next section, and we shall also justify the statement about the choice of the error covariance matrices and why they need to be updated each cycle for the lognormal approach to be optimal.

21.10 Reverse Lognormal Variational Data Assimilation

In Chapters 3 and 4 we introduced the univariate and multivariate versions of the reverse lognormal distribution, along with its mixed combinations with the Gaussian and lognormal distributions. As have seen so far this chapter it is possible from Bayes theorem to derive 3D and 4D VAR based cost functions for the lognormal and mixed Gaussian-lognormal distributions. As such we now present the associated full field 3D and 4D VAR for those different combinations.

21.10.1 3D and 4D Mixed Gaussian-Reverse Lognormal Cost Functions

If we assume that we have a vector that contains $p1$ Gaussian distributed random variables, and $q1$ reverse-lognormal distributed random variables with $\mathbf{T}_{q1}$ upper bounds, along with $p2$ Gaussian distributed observational errors, and $q2$ reverse lognormally distributed observational errors with $\mathbf{T}_{q2}$

upper bounds, then the associated 3D VAR full field cost functions is given by

$$
\begin{aligned}
J_{GR\Lambda}(\mathbf{x}) = & \frac{1}{2}\begin{pmatrix} \mathbf{x}_{p1}^t - \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix}^T \mathbf{B}_{GR\Lambda}^{-1} \begin{pmatrix} \mathbf{x}_{p1}^t - \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \mathbf{x}_{p1}^t - \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p1} \\ \mathbf{1}_{q1} \end{pmatrix} \right\rangle, \\
& + \frac{1}{2}\begin{pmatrix} \mathbf{y}_{p2} - \mathbf{h}_{p2}(\mathbf{x}^t) \\ \ln(\mathbf{T}_{q2} - \mathbf{y}_{q2}) - \ln(\mathbf{T}_{q2} - \mathbf{h}_{q2}(\mathbf{x}^t)) \end{pmatrix}^T \mathbf{R}_{GR\Lambda}^{-1} \begin{pmatrix} \mathbf{y}_{p2} - \mathbf{h}_{p2}(\mathbf{x}^t) \\ \ln(\mathbf{T}_{q2} - \mathbf{y}_{q2}) - \ln(\mathbf{T}_{q2} - \mathbf{h}_{q2}(\mathbf{x}^t)) \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \mathbf{y}_{p2} - \mathbf{h}_{p2}(\mathbf{x}^t) \\ \ln(\mathbf{T}_{q2} - \mathbf{y}_{q2}) - \ln(\mathbf{T}_{q2} - \mathbf{h}_{q2}(\mathbf{x}^t)) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p2} \\ \mathbf{1}_{q2} \end{pmatrix} \right\rangle. \qquad (21.97)
\end{aligned}
$$

The 4D VAR equivalent cost function is given by

$$
\begin{aligned}
& J_{GR\Lambda}(\mathbf{x}_0) \\
& = \frac{1}{2}\begin{pmatrix} \mathbf{x}_{p1,0}^t - \mathbf{x}_{p1,0}^b \\ \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^t\right) - \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^b\right) \end{pmatrix}^T \mathbf{B}_{GR\Lambda}^{-1} \begin{pmatrix} \mathbf{x}_{p1,0}^t - \mathbf{x}_{p1,0}^b \\ \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^t\right) - \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^b\right) \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \mathbf{x}_{p1,0}^t - \mathbf{x}_{p1,0}^b \\ \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^t\right) - \ln\left(\mathbf{T}_{q1,0} - \mathbf{x}_{q1,0}^b\right) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p1} \\ \mathbf{1}_{q1} \end{pmatrix} \right\rangle, \\
& + \frac{1}{2}\sum_{i=1}^{N_o} \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}_i^t) \\ \ln(\mathbf{T}_{q2,i} - \mathbf{y}_{q2,i}) - \ln(\mathbf{T}_{q2,i} - \mathbf{h}_{q2,i}(\mathbf{x}_i^t)) \end{pmatrix}^T \mathbf{R}_{GR\Lambda,i}^{-1} \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}_i^t) \\ \ln(\mathbf{T}_{q2,i} - \mathbf{y}_{q2,i}) - \ln(\mathbf{T}_{q2,i} - \mathbf{h}_{q2,i}(\mathbf{x}_i^t)) \end{pmatrix} \\
& + \sum_{i=1}^{N_o} \left\langle \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}_i^t) \\ \ln(\mathbf{T}_{q2,i} - \mathbf{y}_{q2,i}) - \ln(\mathbf{T}_{q2,i} - \mathbf{h}_{q2,i}(\mathbf{x}_i^t)) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p2,i} \\ \mathbf{1}_{q2,i} \end{pmatrix} \right\rangle, \qquad (21.98)
\end{aligned}
$$

for $i = 1, 2, \ldots, N_0$, and where $\mathbf{x}_i \equiv \mathcal{M}_{0,i}(\mathbf{x}_0)$. We should note there are different $\mathbf{T}$ as each observation time of varying size.

21.10.2 3D and 4D Mixed Lognormal-Reverse Lognormal Cost Functions

If we assume that we have a vector that contains $p1$ lognormally distributed random variables, and $q1$ reverse-lognormal distributed random variables with $\mathbf{T}_{q1}$ upper bounds, along with $p2$ lognormally distributed observational errors, and $q2$ reverse lognormally distributed observational errors with $\mathbf{T}_{q2}$ upper bounds, then the associated 3D VAR full field cost functions is given by

$$
\begin{aligned}
J_{LR\Lambda}(\mathbf{x}) = & \frac{1}{2}\begin{pmatrix} \ln \mathbf{x}_{p1}^t - \ln \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix}^T \mathbf{B}_{LR\Lambda}^{-1} \begin{pmatrix} \ln \mathbf{x}_{p1}^t - \ln \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix} \\
& + \left\langle \begin{pmatrix} \ln \mathbf{x}_{p1}^t - \ln \mathbf{x}_{p1}^b \\ \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^t\right) - \ln\left(\mathbf{T}_{q1} - \mathbf{x}_{q1}^b\right) \end{pmatrix}, \begin{pmatrix} \mathbf{1}_{p1} \\ \mathbf{1}_{q1} \end{pmatrix} \right\rangle,
\end{aligned}
$$

$$+\frac{1}{2}\begin{pmatrix}\ln \mathbf{y}_{p2}-\ln \mathbf{h}_{p2}\left(\mathbf{x}^t\right)\\ \ln\left(\mathbf{T}_{q2}-\mathbf{y}_{q2}\right)-\ln\left(\mathbf{T}_{q2}-\mathbf{h}_{q2}\left(\mathbf{x}^t\right)\right)\end{pmatrix}^T\mathbf{R}_{LR\Lambda}^{-1}\begin{pmatrix}\ln \mathbf{y}_{p2}-\ln \mathbf{h}_{p2}\left(\mathbf{x}^t\right)\\ \ln\left(\mathbf{T}_{q2}-\mathbf{y}_{q2}\right)-\ln\left(\mathbf{T}_{q2}-\mathbf{h}_{q2}\left(\mathbf{x}^t\right)\right)\end{pmatrix}$$
$$+\left\langle\begin{pmatrix}\ln \mathbf{y}_{p2}-\ln \mathbf{h}_{p2}\left(\mathbf{x}^t\right)\\ \ln\left(\mathbf{T}_{q2}-\mathbf{y}_{q2}\right)-\ln\left(\mathbf{T}_{q2}-\mathbf{h}_{q2}\left(\mathbf{x}^t\right)\right)\end{pmatrix},\begin{pmatrix}\mathbf{1}_{p2}\\ \mathbf{1}_{q2}\end{pmatrix}\right\rangle. \tag{21.99}$$

The 4D VAR equivalent cost function is given by

$$\begin{aligned}&J_{LR\Lambda}\left(\mathbf{x}_0\right)\\&=\frac{1}{2}\begin{pmatrix}\ln \mathbf{x}_{p1,0}^t-\ln \mathbf{x}_{p1,0}^b\\ \ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^t\right)-\ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^b\right)\end{pmatrix}^T\mathbf{B}_{LR\Lambda}^{-1}\begin{pmatrix}\ln \mathbf{x}_{p1,0}^t-\ln \mathbf{x}_{p1,0}^b\\ \ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^t\right)-\ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^b\right)\end{pmatrix}\\&+\left\langle\begin{pmatrix}\ln \mathbf{x}_{p1,0}^t-\ln \mathbf{x}_{p1,0}^b\\ \ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^t\right)-\ln\left(\mathbf{T}_{q1,0}-\mathbf{x}_{q1,0}^b\right)\end{pmatrix},\begin{pmatrix}\mathbf{1}_{p1}\\ \mathbf{1}_{q1}\end{pmatrix}\right\rangle,\\&+\frac{1}{2}\sum_{i=1}^{N_o}\begin{pmatrix}\ln \mathbf{y}_{p2,i}-\ln \mathbf{h}_{p2,i}\left(\mathbf{x}_i^t\right)\\ \ln\left(\mathbf{T}_{q2,i}-\mathbf{y}_{q2,i}\right)-\ln\left(\mathbf{T}_{q2,i}-\mathbf{h}_{q2,i}\left(\mathbf{x}_i^t\right)\right)\end{pmatrix}^T\mathbf{R}_{LR\Lambda,i}^{-1}\begin{pmatrix}\ln \mathbf{y}_{p2,i}-\ln \mathbf{h}_{p2,i}\left(\mathbf{x}_i^t\right)\\ \ln\left(\mathbf{T}_{q2,i}-\mathbf{y}_{q2,i}\right)-\ln\left(\mathbf{T}_{q2,i}-\mathbf{h}_{q2,i}\left(\mathbf{x}_i^t\right)\right)\end{pmatrix}\\&+\sum_{i=1}^{N_o}\left\langle\begin{pmatrix}\ln \mathbf{y}_{p2,i}-\ln \mathbf{h}_{p2,i}\left(\mathbf{x}_i^t\right)\\ \ln\left(\mathbf{T}_{q2,i}-\mathbf{y}_{q2,i}\right)-\ln\left(\mathbf{T}_{q2,i}-\mathbf{h}_{q2,i}\left(\mathbf{x}_i^t\right)\right)\end{pmatrix},\begin{pmatrix}\mathbf{1}_{p2,i}\\ \mathbf{1}_{q2,i}\end{pmatrix}\right\rangle,\end{aligned} \tag{21.100}$$

and where $\mathbf{x}_i$ is defined earlier.

21.10.3 3D and 4D Mixed Gaussian-Lognormal-Reverse-Lognormal Cost Functions

If we assume that we have a vector that contains $p1$ Gaussian distributed random variables, $q1$ lognormally distributed random variables, and $r1$ reverse-lognormal distributed random variables with $\mathbf{T}_{r1}$ upper bounds, along with $p2$ Gaussian distributed observational errors, $q2$ lognormally distributed observational errors, and $r2$ reverse lognormally distributed observational errors with $\mathbf{T}_{r2}$ upper bounds, then the associated 3D VAR full field cost functions is given by

$$\begin{aligned}J_{GLR\Lambda}\left(\mathbf{x}\right)&=\frac{1}{2}\begin{pmatrix}\mathbf{x}_{p1}^t-\mathbf{x}_{p1}^b\\ \ln \mathbf{x}_{q1}^t-\ln \mathbf{x}_{q1}^b\\ \ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^t\right)-\ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^b\right)\end{pmatrix}^T\mathbf{B}_{GLR\Lambda}^{-1}\begin{pmatrix}\mathbf{x}_{p1}^t-\mathbf{x}_{p1}^b\\ \ln \mathbf{x}_{q1}^t-\ln \mathbf{x}_{q1}^b\\ \ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^t\right)-\ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^b\right)\end{pmatrix}\\&+\left\langle\begin{pmatrix}\mathbf{x}_{p1}^t-\mathbf{x}_{p1}^b\\ \ln \mathbf{x}_{q1}^t-\ln \mathbf{x}_{q1}^b\\ \ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^t\right)-\ln\left(\mathbf{T}_{r1}-\mathbf{x}_{r1}^b\right)\end{pmatrix},\begin{pmatrix}\mathbf{0}_{p1}\\ \mathbf{1}_{q1}\\ \mathbf{1}_{r1}\end{pmatrix}\right\rangle,\\&+\frac{1}{2}\begin{pmatrix}\mathbf{y}_{p2}-\mathbf{h}_{p2}\left(\mathbf{x}\right)\\ \ln \mathbf{y}_{q2}-\ln \mathbf{h}_{q2}\left(\mathbf{x}^t\right)\\ \ln\left(\mathbf{T}_{r2}-\mathbf{y}_{r2}\right)-\ln\left(\mathbf{T}_{r2}-\mathbf{h}_{r2}\left(\mathbf{x}^t\right)\right)\end{pmatrix}^T\mathbf{R}_{GLR\Lambda}^{-1}\begin{pmatrix}\mathbf{y}_{p2}-\mathbf{h}_{p2}\left(\mathbf{x}\right)\\ \ln \mathbf{y}_{q2}-\ln \mathbf{h}_{q2}\left(\mathbf{x}^t\right)\\ \ln\left(\mathbf{T}_{r2}-\mathbf{y}_{r2}\right)-\ln\left(\mathbf{T}_{r2}-\mathbf{h}_{r2}\left(\mathbf{x}^t\right)\right)\end{pmatrix}\end{aligned}$$

$$+\left\langle \begin{pmatrix} \mathbf{y}_{p2} - \mathbf{h}_{p2}(\mathbf{x}) \\ \ln \mathbf{y}_{q2} - \ln \mathbf{h}_{q2}(\mathbf{x}^t) \\ \ln(\mathbf{T}_{r2} - \mathbf{y}_{r2}) - \ln(\mathbf{T}_{r2} - \mathbf{h}_{r2}(\mathbf{x}^t)) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p2} \\ \mathbf{1}_{q2} \\ \mathbf{1}_{r2} \end{pmatrix} \right\rangle. \tag{21.101}$$

The 4D VAR equivalent cost function is given by

$$\begin{aligned} &J_{GLR\Lambda}(\mathbf{x}_0) \\ &= \frac{1}{2} \begin{pmatrix} \mathbf{x}^t_{p1,0} - \mathbf{x}^b_{p1,0} \\ \ln \mathbf{x}^t_{q1,0} - \ln \mathbf{x}^b_{q1,0} \\ \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^t_{r1,0}\right) - \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^b_{r1,0}\right) \end{pmatrix}^T \mathbf{B}^{-1}_{GLR\Lambda} \begin{pmatrix} \mathbf{x}^t_{p1,0} - \mathbf{x}^b_{p1,0} \\ \ln \mathbf{x}^t_{q1,0} - \ln \mathbf{x}^b_{q1,0} \\ \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^t_{r1,0}\right) - \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^b_{r1,0}\right) \end{pmatrix} \\ &+ \left\langle \begin{pmatrix} \mathbf{x}^t_{p1,0} - \mathbf{x}^b_{p1,0} \\ \ln \mathbf{x}^t_{q1,0} - \ln \mathbf{x}^b_{q1,0} \\ \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^t_{r1,0}\right) - \ln\left(\mathbf{T}_{r1,0} - \mathbf{x}^b_{r1,0}\right) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p1} \\ \mathbf{1}_{q1} \\ \mathbf{1}_{r1} \end{pmatrix} \right\rangle, \\ &+ \frac{1}{2} \sum_{i=1}^{N_o} \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}^t_i) \\ \ln \mathbf{y}_{q2,i} - \ln \mathbf{h}_{q2,i}(\mathbf{x}^t_i) \\ \ln(\mathbf{T}_{r2,i} - \mathbf{y}_{r2,i}) - \ln(\mathbf{T}_{r2,i} - \mathbf{h}_{r2,i}(\mathbf{x}^t_i)) \end{pmatrix}^T \mathbf{R}^{-1}_{GLR\Lambda,i} \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}^t_i) \\ \ln \mathbf{y}_{q2,i} - \ln \mathbf{h}_{q2,i}(\mathbf{x}^t_i) \\ \ln(\mathbf{T}_{r2,i} - \mathbf{y}_{r2,i}) - \ln(\mathbf{T}_{r2,i} - \mathbf{h}_{r2,i}(\mathbf{x}^t_i)) \end{pmatrix} \\ &+ \sum_{i=1}^{N_o} \left\langle \begin{pmatrix} \mathbf{y}_{p2,i} - \mathbf{h}_{p2,i}(\mathbf{x}_i) \\ \ln \mathbf{y}_{q2,i} - \ln \mathbf{h}_{q2,i}(\mathbf{x}^t_i) \\ \ln(\mathbf{T}_{r2,i} - \mathbf{y}_{r2,i}) - \ln(\mathbf{T}_{r2,i} - \mathbf{h}_{r2,i}(\mathbf{x}^t_i)) \end{pmatrix}, \begin{pmatrix} \mathbf{0}_{p2,i} \\ \mathbf{1}_{q2,i} \\ \mathbf{1}_{r2,i} \end{pmatrix} \right\rangle, \end{aligned} \tag{21.102}$$

21.11 Lognormal and Mixed Gaussian-Lognormal Kalman Filters

The derivations presented so far have been for the variational based approach to data assimilation, but as we saw earlier, the ensemble based approaches rely on the Kalman filter equations for their basis. There has been a drive to find a non-Gaussian based approach for the Kalman filter to enable the ensemble filters to have a grounding in a direct relationship to non-Gaussian distributions. In this section we show first that it is not possible to follow the least squares approach shown earlier, but that if we form a cost function for the median of the lognormal distribution then we can derive the mean in $\ln x$ space as well as the analysis and forecast error covariance matrices in the same space, which if we recall, these are the parameters that the distribution is defined with respect to. We start presenting why we cannot follow the least squares approach.

21.11.1 Attempted Derivation at a Lognormal Based Kalman Filter

As just mentioned, a first approach to derive a lognormal based Kalman filter data assimilation system would be to follow the statistical derivation summarized in [130], but with the lognormal equivalent in parallel. The starting point is to define the Gaussian distributed analysis errors as

$$\boldsymbol{\varepsilon}_a \equiv \boldsymbol{x}_a - \boldsymbol{x}_t, \tag{21.103}$$

where $\boldsymbol{x}_t$ is the true states that is being sought, and $\boldsymbol{x}_a$ is the analysis state at the current time. The next step is to introduce the time component, along with the background or forecast states, in terms of the numerical model operating on the analysis state at the previous time step, which results in

$$\boldsymbol{\varepsilon}_f \equiv \boldsymbol{M}_{n,n-1}\boldsymbol{x}_a^{n-1} - \boldsymbol{x}_t^n, \tag{21.104}$$

where $\boldsymbol{M}_{n,n-1}$ is a linear or linearized numerical model matrix that operates from time step t^{n-1} to time step t^n.

The next step is to introduce the lognormal equivalent of (21.103) and (21.104), which are given by

$$\boldsymbol{\varepsilon}_{al} \equiv \boldsymbol{x}_{al} \oslash \boldsymbol{x}_t, \tag{21.105}$$

$$\ln \boldsymbol{\varepsilon}_{al} \equiv \ln\left(\boldsymbol{M}_{n,n-1}\boldsymbol{x}_{al}^{n-1}\right) - \ln \boldsymbol{x}_t^n, \tag{21.106}$$

where $\oslash$ is the Hadamard division operator, which is a componentwise division operator, and the logarithm in (21.106) is also applied componentwise. As we are assuming that the components of the analysis and true state are lognormally distributed, then we have the property that all of these entries are greater than zero. The subscript l above, and throughout, is referring to the components associated with the lognormal formulations. The reason that the analysis error in (21.105) is defined as a ratio is as a result of the lognormal distribution being a geometric distribution, which implies that the ratio, or product, of two independent lognormal random variables is itself a lognormal random variable, whereas for the Gaussian distribution the equivalent property is that for two independent Gaussian random variables, their sum, or difference, is also a Gaussian random variable. This is then implying that we are assuming that if all of the background and observational errors are lognormally distributed, then we are assuming that the analysis error is also a lognormally distributed random variable, which is the equivalent to the assumption that is made for the Gaussian approach.

The next step is to define the analysis states at time step t^{n-1}, which for the Gaussian and lognormal approaches are given by

$$\boldsymbol{x}_a^{n-1} = \boldsymbol{x}_t^{n-1} + \boldsymbol{\varepsilon}_a^{n-1}, \tag{21.107}$$

$$\boldsymbol{x}_{al}^{n-1} = \boldsymbol{x}_t^{n-1} \odot \boldsymbol{\varepsilon}_{al}^{n-1}, \tag{21.108}$$

respectively, where $\odot$ is the Hadamard product operator, which is a componentwise multiplication operator.

Given the analysis error definitions above, the next step in the derivation of the Kalman filter is to introduce the definitions for the forecast, or background, errors which are given by

$$\boldsymbol{\varepsilon}_b = \boldsymbol{M}\boldsymbol{x}_t^{n-1} + \boldsymbol{M}\boldsymbol{\varepsilon}_a^{n-1} - \boldsymbol{x}_t^n, \tag{21.109}$$

$$\ln \boldsymbol{\varepsilon}_{bl} = \ln \boldsymbol{M}\boldsymbol{x}_t^{n-1} + \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al}^{n-1} - \ln \boldsymbol{x}_t^n. \tag{21.110}$$

The reason for presenting the logarithm of the background error term in (21.110) is due to how the Kalman filter equations are derived for the Gaussian approach from their associated expectations definitions. The lognormal distribution is defined in terms of expectations of the Gaussian random variable $\ln \boldsymbol{x}$ and not the lognormally distributed random variable $\boldsymbol{x}$, and as such this is to ensure consistency with the lognormal distribution the errors need to be defined as above.

Given these definitions, it is known that at the previous filtering time, the associated analysis state, denoted with the subscript a, at that time step has an analysis error such that

$$\boldsymbol{x}_a^{n-1} = \boldsymbol{x}_t^{n-1} + \boldsymbol{\varepsilon}_a^{n-1}. \tag{21.111}$$

Where for the lognormal approach we would like something of the form

$$\ln \boldsymbol{x}_{al}^{n-1} = \ln \boldsymbol{x}_t^{n-1} + \ln \boldsymbol{\varepsilon}_{al}^{n-1}, \tag{21.112}$$

for the lognormal based formulation. This then makes it possible to factorize (21.109) such that

$$\boldsymbol{\varepsilon}_b = \boldsymbol{M}\boldsymbol{\varepsilon}_a^{n-1} + \boldsymbol{\varepsilon}_m^n, \tag{21.113}$$

where b denotes the background state, and

$$\boldsymbol{\varepsilon}_m^n \equiv \boldsymbol{M}\boldsymbol{x}_t^{n-1} - \boldsymbol{x}_t^n, \tag{21.114}$$

is the model error term.

However, while it is desirable for the lognormal version of the time evolution of the lognormal analysis error to be as presented in (21.110), unfortunately this is incorrect. This is unfortunately due to the linear model not being able to commute through the Hadamard product operator, and as such another definition is required for the time evolution of the lognormal analysis error. A possible work around is to understand that the background error at the next analysis time $t = n$ can be defined as the true state at that analysis time being multiplied by the evolution of the analysis error from the previous analysis time, but to keep all the terms consistent we have to multiply by the lognormal model error. Thus we would have

$$\ln \boldsymbol{\varepsilon}_b = \ln \boldsymbol{x}_t^n + \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al}^{n-1} + \ln \boldsymbol{\varepsilon}_{ml} - \ln \boldsymbol{x}_t^n = \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al}^{n-1} + \ln \boldsymbol{\varepsilon}_{ml}, \tag{21.115}$$

where $\boldsymbol{\varepsilon}_{ml}$ is defined as

$$\ln \boldsymbol{\varepsilon}_{ml} \equiv \ln \boldsymbol{M}\boldsymbol{x}_t^{n-1} - \ln \boldsymbol{x}_t^n. \tag{21.116}$$

This implies that it is not possible to move the numerical model out of the logarithm. This causes a problem as it is not the evolution of the logarithm of the analysis errors that is needed, but rather the logarithm of the evolution of analysis error.

The next step in the derivation of the Kalman filter equations is to form the forecast error covariance matrix which comes from multiplying (21.115) with its transpose and then applying the statistical expectation operator, $\mathbb{E}[\cdot]$, to this product, which yields

$$\begin{aligned}
\boldsymbol{P}_b^n &\equiv \mathbb{E}\left[\boldsymbol{\varepsilon}_b^n \left(\boldsymbol{\varepsilon}_b^n\right)^T\right] = \mathbb{E}\left[\left(\boldsymbol{M}\boldsymbol{\varepsilon}_a^{n-1} + \boldsymbol{\varepsilon}_m^n\right)\left(\boldsymbol{M}\boldsymbol{\varepsilon}_a^{n-1} + \boldsymbol{\varepsilon}_m^n\right)^T\right], \\
&= \boldsymbol{M}\mathbb{E}\left[\boldsymbol{\varepsilon}_a^{n-1}\left(\boldsymbol{\varepsilon}_a^{n-1}\right)^T\right]\boldsymbol{M}^T + \mathbb{E}\left[\boldsymbol{\varepsilon}_m^n \left(\boldsymbol{\varepsilon}_m^n\right)^T\right], \\
&= \boldsymbol{M}\boldsymbol{P}_a^{n-1}\boldsymbol{M}^T + \boldsymbol{Q}^n,
\end{aligned} \tag{21.117}$$

where $\boldsymbol{P}_a$ is the analysis error covariance matrix, and $\boldsymbol{Q}$ is the model error covariance matrix, and we have assumed that the analysis error and the model error are uncorrelated.

Applying the same approach to (21.115) does yield a lognormal forecast error covariance matrix and a lognormal based model error covariance matrix as

$$\begin{aligned}\boldsymbol{P}_{fl}^{n} &\equiv \mathbb{E}\left[\left(\ln \boldsymbol{M}\boldsymbol{\varepsilon}_{a}^{n-1} + \ln \boldsymbol{\varepsilon}_{m}^{n}\right)\left(\ln \boldsymbol{M}\boldsymbol{\varepsilon}_{a}^{n-1} + \ln \boldsymbol{\varepsilon}_{m}^{n}\right)^{T}\right], \\ &= \mathbb{E}\left[\ln \boldsymbol{M}\boldsymbol{\varepsilon}_{a}^{n-1}\left(\ln \boldsymbol{M}\boldsymbol{\varepsilon}_{a}^{n-1}\right)^{T}\right] + \mathbb{E}\left[\ln \boldsymbol{\varepsilon}_{m}^{n}\left(\ln \boldsymbol{\varepsilon}_{m}^{n}\right)^{T}\right], \\ &= \boldsymbol{P}_{al}^{n} + \boldsymbol{Q}_{l}^{n}, \end{aligned} \tag{21.118}$$

and have again assumed that the analysis and model errors are uncorrelated. It is clear from (21.118) that the definition for the forecast error covariance matrix does not explicitly contain the numerical model acting on the analysis error covariance matrix from the previous analysis time. It is, however, implicit in the state $\boldsymbol{x}^{n}$. It should be noted here that given that it is not possible to interchange the model and the logarithm, for the remainder of the paper the new approach will be with the nonlinear model.

Another problem with trying to derive a lognormal version of the Kalman filter arises from considering the analysis step, as shown in [130], where if given a predicted state $\boldsymbol{x}^{n+1|n+1}$ that is associated with observations up to time step t^{n}, and assuming that an observation has been received at time $t = t^{n+1}$, then an estimate of the state at $t = t^{n+1}$, given the observation at time $t = t^{n+1}$ is required. In Kalman filter theory this step is started by assuming that the estimate is a weighted sum of the predicted state and the new observation, that is to say

$$\boldsymbol{x}^{n+1|n+1} = \boldsymbol{K}_{b}^{n+1}\boldsymbol{x}^{n+1|n} + \boldsymbol{K}_{o}^{n+1}\boldsymbol{y}^{n+1}, \tag{21.119}$$

where $\boldsymbol{y}^{n+1}$ is the observation at time $t = t^{n+1}$. We should note here that the observation could be either a direct or indirect observation of the predicted state. However, for a lognormal approach the equivalence of (21.119) would be in terms of a weighted sum of the logarithm of the predicted state and the observations such that

$$\ln \boldsymbol{x}^{n+1|n+1} = \ln \boldsymbol{K}_{bl}^{n+1}\boldsymbol{x}^{n+1|n} + \ln \boldsymbol{K}_{ol}^{n+1}\boldsymbol{y}^{n+1}, \tag{21.120}$$

where the K matrices in (21.119) and (21.120) are referred to as gain matrices. Thus it is not possible to manipulate the equations to obtain the expressions for the gain matrices due to the logarithms being present and acting on the state. Therefore it is not possible to interchange the operators. Thus this approach cannot continue through the steps of the derivation of the Gaussian based Kalman filter equations to seek an equivalent lognormal version of the Kalman gain matrix and filter equations.

An approach proposed to obtain a form of a lognormal based Kalman filter in [223] was based upon recasting the deterministic model in terms of log of the variable of interest and then running the Kalman filter on with this model, and then converting back to the lognormal space random variable. This is not practical for an atmospheric or oceanic numerical model. Thus we now present a new approach to derive the equations for a lognormal and mixed lognormal-Gaussian based Kalman filter from [139].

21.11.2 Lognormal Kalman Filter - Median Based Approach

It appears from the attempted derivation in Section 5.54 that it is not possible to obtain a lognormal version of the Kalman filter equations, that are based upon the first two moments of the multivariate Gaussian distribution through following a least squares approaches. However, what is important to recall here is that for the Gaussian distribution the three descriptive statistics are the same, that is to say the mode, median, and the mean are the same. This is not true for the lognormal distribution. It is shown in [135] that these three statistics are quite different, and that they each have their own properties; the

mode is the maximum likelihood state, the median is the unbiased state, and the mean is the minimum variance state.

Given these properties, it is quite often stated that an error is unbiased if its mean is equal to zero. This is not true for many non-Gaussian distributed random variables. As just stated, the median is the unbiased state, and as such this implies that the cumulative density function, when integrated to this value, is equal to 0.5. For example to say that a lognormal random variable, or error, is unbiased if $\mathbb{E}[\varepsilon] = 0$, is equivalent to saying that

$$\exp\left\{\mu + \frac{\sigma^2}{2}\right\} = 0, \tag{21.121}$$

where $\mu = \mathbb{E}[\ln \varepsilon]$ and σ^2 is the variance of $\ln \varepsilon$ and not ε, which can not happen. It is not possible to have a zero mean for a lognormal random variable; however, it is possible that $\mathbb{E}[\ln \varepsilon] = 0$, but this implies that the distributions of the true state and the background state are the same type and that they have the same median, not the same mean, in lognormal space, which is also true in the Gaussian transformed space as well.

To help illustrate this point, we have plotted two lognormal distributions that are supposed to represent the probability density functions (PDFs) of the true state, solid curve, and the analysis state, dashed curve, where the two states have the same Gaussian mean, $\mu_t = \mu_a = \ln 2$, but with different Gaussian variances, 0.25^2 and 0.5^2 respectively, in Fig. 21.12. We have also plotted lines where the median, modes, and means for the two distributions in this order with their respective solid or dashed line associated with the solid of dashed curve.

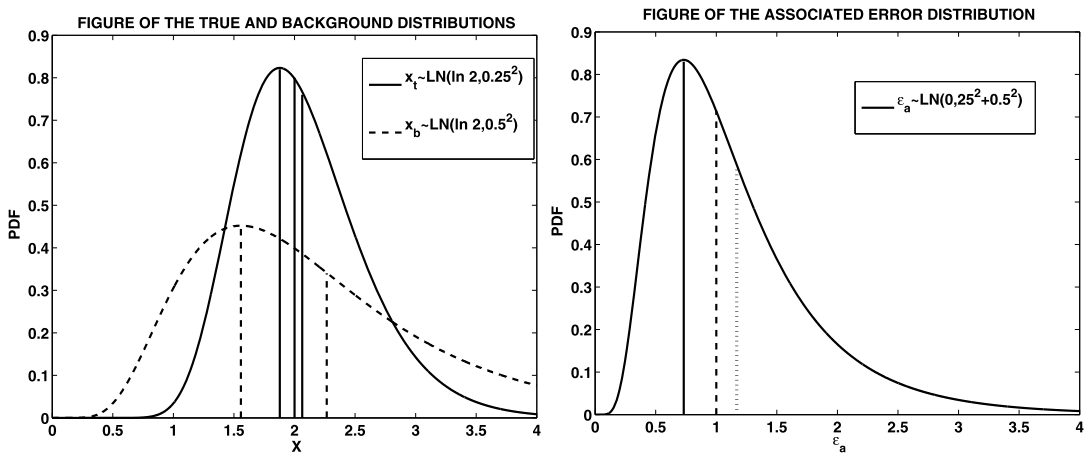


FIGURE 21.12

Plots to illustrate the differences in the modes, medians, and means for two different lognormal distributions that are representing the true state's distribution (solid curve), and the analysis state's distribution (dashed curve) in the left plot, whilst the plot on the right is showing the distribution for the associated analysis error $\varepsilon_a = \dfrac{x_a}{x_t}$.

It is clear from the left plot in Fig. 21.12 that the two distributions have the same median, but that neither their modes nor their means are equal. In the right plot in Fig. 21.12 we have plotted the associated distribution for the equivalent analysis error as $\varepsilon_a = \dfrac{x_a}{x_t}$. While the distribution has a median at 1, the most likely state is to the left of this value, indicating that the state with the highest probability of occurring for the analysis error is not equality, which implies a bias in the analysis.

In [132] it is shown that when following a modal approach for the incremental formulation of mixed lognormal-Gaussian 4DVAR, the analysis error distribution, or the posterior distribution as it is also referred to as, had a mode at 1. This is therefore indicating that the most likely answer from the data assimilation system was something close to the true state.

Given this brief explanation and illustrations of the descriptive statistics of the lognormal distribution, it is clear that the way to derive an equivalent set of Kalman filter type equations for lognormally distributed errors, using the forecast error covariance matrix evolution described earlier, is to start with defining a cost function who's solution is the median of the analysis error distribution [130], as this is equivalent to $\ln \boldsymbol{\varepsilon}_a$ and then find the associated covariance matrix for $\ln \boldsymbol{\varepsilon}_a$. In [130] it is shown that the median analysis state for lognormally distributed background, and observation, errors is the minimum of

$$J(\boldsymbol{x}) = \frac{1}{2}(\ln \boldsymbol{x}_t - \ln \boldsymbol{x}_b)^T \boldsymbol{P}_{fl}^{-1}(\ln \boldsymbol{x}_t - \ln \boldsymbol{x}_b) + \frac{1}{2}(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_t))^T \boldsymbol{R}_l^{-1}(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_t)). \tag{21.122}$$

Differentiating (21.122) with respect to $\boldsymbol{x}_t$ results in

$$\nabla_{\boldsymbol{x}_t} J(\boldsymbol{x}) = \boldsymbol{W}_b^{-T} \boldsymbol{P}_{fl}^{-1}(\ln \boldsymbol{x}_t - \ln \boldsymbol{x}_b) - \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1}(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_t)), \tag{21.123}$$

where

$$\boldsymbol{W}_b^{-1} = \begin{pmatrix} x_{1t}^{-1} & 0 & 0 & \cdots & 0 \\ 0 & x_{2t}^{-1} & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & x_{nt}^{-1} \end{pmatrix} \quad \boldsymbol{W}_o^{-1} = \begin{pmatrix} (\boldsymbol{h}_1(\boldsymbol{x}_t))^{-1} & 0 & 0 & \cdots & 0 \\ 0 & (\boldsymbol{h}_2(\boldsymbol{x}_t))^{-1} & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & (\boldsymbol{h}_{N_o}(\boldsymbol{x}_t))^{-1} \end{pmatrix}. \tag{21.124}$$

As it will be required later to finalize the derivation of the lognormal Kalman filter, it can be shown that the scaled Hessian matrix of (21.122) is

$$\begin{aligned} \nabla_{\boldsymbol{x}_t}^2 J(\boldsymbol{x}) &= \boldsymbol{W}_b^{-T} \boldsymbol{P}_{fl}^{-1} \boldsymbol{W}_b^{-1} + \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H}, \\ \boldsymbol{W}_b^T \nabla_{\boldsymbol{x}_t}^2 J(\boldsymbol{x}) \boldsymbol{W}_b &= \boldsymbol{P}_{fl}^{-1} + \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b. \end{aligned} \tag{21.125}$$

An important feature to notice about (21.125) is that the Hessian matrix is positive definite due to the forecast error covariance matrix being positive definite, and while the second term may not appear obviously positive definite, it is due the fact that it can be factorized into the product of a matrix and its transpose which results in the matrix of the product being positive definite.

As with minimization of the Gaussian cost-function in 3DVAR, setting the Jacobian in (21.123) equal to zero, and calling the state that achieves this the analysis state $\boldsymbol{x}_a$, yields

$$\nabla_{\boldsymbol{x}_t}(\boldsymbol{x}_a) = \boldsymbol{W}_b^{-T}\boldsymbol{P}_{fl}^{-1}(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b) - \boldsymbol{H}^T\boldsymbol{W}_o^{-T}\boldsymbol{R}_l^{-1}(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_a)) = 0. \tag{21.126}$$

The next step in deriving the lognormal version of the Kalman filter equations is to state that we wish to associate the analysis state with the observations in terms of some form of lognormal based Kalman gain matrix, $\boldsymbol{K}_l$. Thus we state that we require

$$\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b = \boldsymbol{K}_l(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_a)). \tag{21.127}$$

The reason for the form in (21.127) is so that when we have to take the expectation to derive the analysis error covariance matrix, it is assumed to be lognormal, and as such is in terms of the logarithm of the random variable, and not the random variable itself.

Next we introduce the logarithmic geometric tangent linear approximation [132], which enables the logarithm of the model fields to be operated on by the Jacobian of the observation operator $\boldsymbol{h}(\boldsymbol{x})$. The starting point is to consider the numerical geometric derivative of $\ln \boldsymbol{h}(\boldsymbol{x})$, which is given by

$$\left[\frac{\ln \boldsymbol{h}(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{h}(\boldsymbol{x})}{\boldsymbol{x} \odot (\boldsymbol{p}_i - \boldsymbol{1})}\right]_j \approx \left[\boldsymbol{W}_o^{-1}\boldsymbol{H}\right]_j, \quad j = 1, 2, \ldots, N \tag{21.128}$$

where

$$\boldsymbol{W}_{o,jj}^{-1} \equiv \operatorname{diag}\left\{\frac{1}{\boldsymbol{h}_i(\boldsymbol{x})}\right\}, \quad i = 1, 2, \ldots, N_o, \, j = 1, 2, \ldots, N. \tag{21.129}$$

The next step is to multiply, and divide, (21.128) by $\left[\ln(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{x}\right]_j$ and interchange the denominators on the left hand side, which yields

$$\left[\frac{\ln \boldsymbol{h}(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{h}(\boldsymbol{x})}{\ln(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{x}}\right]_j \left[\frac{\ln(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{x}}{\boldsymbol{x} \odot (\boldsymbol{p}_i - \boldsymbol{1})}\right]_j \approx \left[\boldsymbol{W}_o^{-1}\boldsymbol{H}\right]_j. \tag{21.130}$$

Wich is equivalent to

$$\left[\frac{\ln \boldsymbol{h}(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{h}(\boldsymbol{x})}{\ln(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{x}}\boldsymbol{W}_b^{-1}\right]_j \approx \left[\boldsymbol{W}_o^{-1}\boldsymbol{H}\right]_j. \tag{21.131}$$

Where, after some rearrangement, and noticing that the componentwise form above can be expressed in terms of matrirx-vector formulation, yields

$$\ln \boldsymbol{h}(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{h}(\boldsymbol{x}) \approx \boldsymbol{W}_o^{-1}\boldsymbol{H}\boldsymbol{W}_b\left(\ln(\boldsymbol{x} \odot \boldsymbol{p}_i) - \ln \boldsymbol{x}\right). \tag{21.132}$$

Thus we have that

$$\ln \boldsymbol{h}(\boldsymbol{x}_a) \approx \ln \boldsymbol{h}(\boldsymbol{x}_b) + \boldsymbol{W}_o^{-1}\boldsymbol{H}\boldsymbol{W}_b(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b), \tag{21.133}$$

which results in

$$\boldsymbol{W}_b^{-T}\boldsymbol{P}_{fl}^{-1}(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b) - \boldsymbol{W}_o^{-T}\boldsymbol{H}^T\boldsymbol{R}_l^{-1}\left(\ln \boldsymbol{y} - \ln \boldsymbol{h}(\boldsymbol{x}_b) - \boldsymbol{W}_o^{-1}\boldsymbol{H}\boldsymbol{W}_b(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b)\right) = 0. \tag{21.134}$$

The next steps are to pre-multiply (21.134) by $\boldsymbol{W}^T$ and then add, and subtract,

$$\boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b \left(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b\right),$$

which yields

$$\left[\boldsymbol{P}_{fl}^{-1} + \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right] \left(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b\right) = \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \left(\ln \boldsymbol{y} - \ln \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right). \tag{21.135}$$

Upon some rearrangements of (21.135), it is possible to obtain a format as desired in (21.127), given by

$$\left(\ln \boldsymbol{x}_a - \ln \boldsymbol{x}_b\right) = \left[\boldsymbol{P}_{fl}^{-1} + \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right]^{-1} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \left(\ln \boldsymbol{y} - \ln \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right), \tag{21.136}$$

and as such the lognormal Kalman gain matrix is of a similar form to the Gaussian version but now containing the derivatives of the logarithms, and is given by

$$\begin{aligned} \boldsymbol{K}_l &\equiv \left[\boldsymbol{P}_{fl}^{-1} + \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right]^{-1} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \\ &\equiv \boldsymbol{P}_{fl} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \left[\boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b \boldsymbol{P}_{fl} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} + \boldsymbol{R}_l\right]^{-1}. \end{aligned} \tag{21.137}$$

It is the latter expression of the lognormal Kalman gain matrix, $\boldsymbol{K}_l$, that will be used in the next step; which is to derive the lognormal equivalent of the analysis error covariance matrix.

The starting point is to define the analysis error which for a lognormal distribution this is given by

$$\ln \boldsymbol{\varepsilon}_a \equiv \ln \boldsymbol{x}_a - \ln \boldsymbol{x}_t. \tag{21.138}$$

The next step is to substitute for $\ln \boldsymbol{x}_a$ in (21.138) with the expression from (21.136), but with $\boldsymbol{K}_l$ to make things easier to follow. Thus we have

$$\ln \boldsymbol{\varepsilon}_a = \ln \boldsymbol{x}_b - \ln \boldsymbol{x}_t + \boldsymbol{K}_l \left(\ln \boldsymbol{y} - \ln \boldsymbol{h}\left(\boldsymbol{x}_b\right)\right). \tag{21.139}$$

However, to obtain a similar form to that derivation for the Gaussian approach we must rewrite $\ln \boldsymbol{H}\left(\boldsymbol{x}_b\right)$ in terms of the true state and the background error. This is achieved through $\boldsymbol{x}_b = \boldsymbol{x}_t \odot \boldsymbol{\varepsilon}_{ob}$, using the geometric tangent linear approximation, and the logarithmic tangent linear model, which results in

$$\ln \boldsymbol{h}\left(\boldsymbol{x}_b\right) \approx \ln \boldsymbol{h}\left(\boldsymbol{x}_t\right) + \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b \ln \boldsymbol{\varepsilon}_{ob}. \tag{21.140}$$

Substituting (21.140) into (21.139) yields

$$\begin{aligned} \ln \boldsymbol{\varepsilon}_a &= \ln \boldsymbol{\varepsilon}_{ob} + \boldsymbol{K}_l \left(\ln \boldsymbol{y} - \ln \boldsymbol{h}\left(\boldsymbol{x}_t\right) - \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b \ln \boldsymbol{\varepsilon}_{ob}\right), \\ &= \left(\boldsymbol{I} - \boldsymbol{K}_l \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right) \ln \boldsymbol{\varepsilon}_{ob} + \boldsymbol{K}_l \ln \boldsymbol{\varepsilon}_{ol}. \end{aligned} \tag{21.141}$$

To simplify the appearance of the derivation we now define $\widehat{\boldsymbol{H}} \equiv \boldsymbol{W}_b \boldsymbol{H} \boldsymbol{W}_o^{-1}$. To form the lognormal analysis error covariance matrix we have to take the expectation of $\ln \boldsymbol{\varepsilon}_a \left(\ln \boldsymbol{\varepsilon}_a\right)^T$, which is

$$\begin{aligned}\boldsymbol{P}_{al} &\equiv \mathbb{E}\left[\ln \boldsymbol{\varepsilon}_a \left(\ln \boldsymbol{\varepsilon}_a\right)^T\right] \\ &= \left(\boldsymbol{I} - \boldsymbol{K}_l \widehat{\boldsymbol{H}}\right) \mathbb{E}\left[\ln \boldsymbol{\varepsilon}_{ob} \left(\ln \boldsymbol{\varepsilon}_{ob}\right)^T\right] \left(\boldsymbol{I} - \boldsymbol{K}_l \widehat{\boldsymbol{H}}\right)^T + \boldsymbol{K}_l \mathbb{E}\left[\ln \boldsymbol{\varepsilon}_{ol} \left(\ln \boldsymbol{\varepsilon}_{ol}\right)^T\right] \boldsymbol{K}_l^T \\ &= \left(\boldsymbol{I} - \boldsymbol{K}_l \widehat{\boldsymbol{H}}\right) \boldsymbol{P}_{fl} \left(\boldsymbol{I} - \boldsymbol{K}_l \widehat{\boldsymbol{H}}\right)^T + \boldsymbol{K}_l \boldsymbol{R}_l \boldsymbol{K}_l^T. \end{aligned} \tag{21.142}$$

Now following the same expansion of the products in (21.142) as for the Gaussian case, and noticing that the lognormal Kalman gain equation can be written as

$$\boldsymbol{K}_l = \boldsymbol{P}_{fl} \widehat{\boldsymbol{H}} \left[\widehat{\boldsymbol{H}} \boldsymbol{P}_{fl} \widehat{\boldsymbol{H}}^T + \boldsymbol{R}_l\right],$$

then it is possible to write (21.142) as

$$\boldsymbol{P}_{al} = \left(\boldsymbol{I} - \boldsymbol{K}_l \widehat{\boldsymbol{H}}\right) \boldsymbol{P}_{fl}. \tag{21.143}$$

The final proof that is required is in regards to showing that the inverse analysis error covariance matrix is equivalent to the inverse of the Hessian of (21.122). The first step is to expand $\boldsymbol{K}_l$ in (21.143) in terms of $\widehat{\boldsymbol{H}}$, and using the rule for the inverse of the product of two matrices, yields

$$\boldsymbol{P}_{al} = \boldsymbol{P}_{fl} - \boldsymbol{P}_{fl} \widehat{\boldsymbol{H}}^T \boldsymbol{R}_l^{-1} \left[\widehat{\boldsymbol{H}} \boldsymbol{P}_{fl} \widehat{\boldsymbol{H}}^T \boldsymbol{R}_l^{-1} + \boldsymbol{I}\right]^{-1} \widehat{\boldsymbol{H}} \boldsymbol{P}_{fl}. \tag{21.144}$$

Next recalling the Sherman-Morrison-Woodbury formula

$$\left(\boldsymbol{A} + \boldsymbol{U}\boldsymbol{V}^T\right)^{-1} \equiv \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{U}\left[\boldsymbol{I} + \boldsymbol{V}^T\boldsymbol{A}^{-1}\boldsymbol{U}\right]^{-1} \boldsymbol{V}^T\boldsymbol{A}^{-1},$$

where for (21.144) this implies that $\boldsymbol{A} = \boldsymbol{P}_{fl}^{-1}$, $\boldsymbol{U} = \widehat{\boldsymbol{H}}^T \boldsymbol{R}_l^{-1}$ and $\boldsymbol{V} = \widehat{\boldsymbol{H}}^T$, which leads to the lognormal analysis error covariance matrix being defined as

$$\begin{aligned}\boldsymbol{P}_{al} &= \left(\boldsymbol{P}_{fl}^{-1} + \widehat{\boldsymbol{H}}^T \boldsymbol{R}_l^{-1} \widehat{\boldsymbol{H}}\right)^{-1}, \\ &= \left(\boldsymbol{P}_{fl}^{-1} + \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \boldsymbol{R}_l^{-1} \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right)^{-1}, \end{aligned} \tag{21.145}$$

where the expression inside the brackets on the right hand side of (21.145) has already been shown to be the Hessian matrix for (21.122) in (21.125).

Thus in summary the equations for the analysis step of the lognormal Kalman filter are given by

$$\boldsymbol{P}_{fl}^n = \left(\ln M\left(\boldsymbol{x}_b^{n-1} \odot \boldsymbol{\varepsilon}_a^{n-1}\right) - \ln \boldsymbol{x}_t^n\right)\left(\ln M\left(\boldsymbol{x}_b^{n-1} \odot \boldsymbol{\varepsilon}_a^{n-1}\right) - \ln \boldsymbol{x}_t^n\right)^T + \boldsymbol{Q}_l, \tag{21.146}$$

$$\ln \boldsymbol{x}_a = \ln \boldsymbol{x}_b + \boldsymbol{K}_l \left(\ln \boldsymbol{y} - \ln \boldsymbol{h}\left(\boldsymbol{x}_a\right)\right), \tag{21.147}$$

$$\boldsymbol{K}_l = \boldsymbol{P}_{fl} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} \left[\boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b \boldsymbol{P}_{fl} \boldsymbol{W}_b^T \boldsymbol{H}^T \boldsymbol{W}_o^{-T} + \boldsymbol{R}_l\right]^{-1}, \tag{21.148}$$

$$\boldsymbol{P}_{al}^n = \left(\boldsymbol{I} + \boldsymbol{K}_l \boldsymbol{W}_o^{-1} \boldsymbol{H} \boldsymbol{W}_b\right) \boldsymbol{P}_{fl}^n. \tag{21.149}$$

However, as was shown with the development of the lognormal forms of variational data assimilation, we do not live in an one type of distribution only world, and as such the next step is to combine the lognormal theory just shown for the Kalman filter with the Gaussian Kalman filter theory to be able to use the mixed Gaussian-lognormal distribution from [136] to form a second non-Gaussian based Kalman filter system.

21.11.3 Mixed Gaussian-Lognormal Kalman Filter (MXKF)

In this section we shall refer to the mixed Gaussian-lognormal Kalman Filter as MXKF. The starting point for the derivation of the MXKF is the definition of the associated background, observational, model, and analysis errors. As we are assuming that the error that is to minimized is from a mixed distribution, it implies that there are a set of Gaussian distributed errors and a set of lognormal distributed errors that need to be minimized simultaneously. This then implies that the true state, background, and analysis states are given by

$$\boldsymbol{x}_{tmx} \equiv \begin{pmatrix} \boldsymbol{x}_{tG} \\ \boldsymbol{x}_{tl} \end{pmatrix}, \qquad \boldsymbol{x}_{bmx} \equiv \begin{pmatrix} \boldsymbol{x}_{bG} \\ \boldsymbol{x}_{bl} \end{pmatrix} m \qquad \boldsymbol{x}_{amx} \equiv \begin{pmatrix} \boldsymbol{x}_{aG} \\ \boldsymbol{x}_{al}, \end{pmatrix} \tag{21.150}$$

where G represents the Gaussian distributed random variables, and L the lognormally distributed random, which implies that the associated mixed distributed errors are given by

$$\begin{aligned} \boldsymbol{\varepsilon}_{bmx} &\equiv \begin{pmatrix} \boldsymbol{x}_{bG} - \boldsymbol{x}_{tG} \\ \ln \boldsymbol{x}_{bl} - \ln \boldsymbol{x}_{tl} \end{pmatrix}, \quad \boldsymbol{\varepsilon}_{amx} \equiv \begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{tG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{tl} \end{pmatrix}, \\ \boldsymbol{\varepsilon}_{omx} &\equiv \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{H}_G(\boldsymbol{x}_{tmx}) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{H}_l(\boldsymbol{x}_{tmx}) \end{pmatrix}, \quad \boldsymbol{\varepsilon}_{mmx} \equiv \begin{pmatrix} \left(\boldsymbol{M}\boldsymbol{x}_t^{n-1}\right)_G - \boldsymbol{x}_{tG}^n \\ \ln\left(\boldsymbol{M}\boldsymbol{x}_t^{n-1}\right)_l - \ln \boldsymbol{x}_{tl}^n. \end{pmatrix} \end{aligned} \tag{21.151}$$

It should be noted here that the number of Gaussian observation errors will, in most circumstances, not be the same as the true state. This is also true for the lognormal distributed errors. Finally there may not be an equal number of Gaussian and lognormal background, or observational, errors.

The mixed distribution based forecast error covariance matrix can be shown to be

$$\boldsymbol{P}_{fmx} \equiv \begin{pmatrix} \boldsymbol{M}\boldsymbol{\varepsilon}_{aG} \\ \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al} \end{pmatrix} \begin{pmatrix} \boldsymbol{M}\boldsymbol{\varepsilon}_{aG} \\ \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al} \end{pmatrix}^T \tag{21.152}$$

where it can clearly be seen that there are covariances between the Gaussian and the lognormal forecast errors.

The next step is to define the equivalent cost function from the lognormal median approach to find the median of the mixed distribution, which is given by

$$\begin{aligned} J_{mx}(\boldsymbol{x}) = &\frac{1}{2} \begin{pmatrix} \boldsymbol{x}_{tG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{tl} - \ln \boldsymbol{x}_{bl} \end{pmatrix}^T \boldsymbol{P}_{fmx}^{-1} \begin{pmatrix} \boldsymbol{x}_{tG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{tl} - \ln \boldsymbol{x}_{bl} \end{pmatrix} \\ &+ \frac{1}{2} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{H}_G(\boldsymbol{x}_t) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{H}_l(\boldsymbol{x}_t) \end{pmatrix}^T \boldsymbol{R}_{mx}^{-1} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{H}_G(\boldsymbol{x}_t) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{H}_l(\boldsymbol{x}_t) \end{pmatrix}, \end{aligned} \tag{21.153}$$

where the Jacobian of (21.153) can be shown to be

$$\nabla_{\boldsymbol{x}_t} J(\boldsymbol{x}) = \widetilde{\boldsymbol{W}}_b^{-T} \boldsymbol{P}_{fmx}^{-1} \begin{pmatrix} \boldsymbol{x}_{tG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{tl} - \ln \boldsymbol{x}_{bl} \end{pmatrix} - \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{H}_G(\boldsymbol{x}_t) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{H}_l(\boldsymbol{x}_t) \end{pmatrix}, \tag{21.154}$$

and the associated Hessian matrix is given by

$$\nabla_{\boldsymbol{x}_t}^2 J(\boldsymbol{x}) = \widetilde{\boldsymbol{W}}_b^{-T} \boldsymbol{P}_{fmx}^{-1} \widetilde{\boldsymbol{W}}_b^{-1} + \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_0^{-1} \boldsymbol{H} \equiv \boldsymbol{P}_{fmx}^{-1} + \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_0^{-1} \boldsymbol{H} \widetilde{\boldsymbol{W}}_b, \tag{21.155}$$

where

$$\widehat{\boldsymbol{W}}_b^{-1} \equiv \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & x_{tl_1}^{-1} & & \\ & & & & \ddots & \\ & & & & & x_{tl_N}^{-1} \end{pmatrix}, \quad \widehat{\boldsymbol{W}}_o^{-1} \equiv \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & & \boldsymbol{h}_{l_1}(\boldsymbol{x}_t) & & \\ & & & & \ddots & \\ & & & & & \boldsymbol{h}_{l_N}(\boldsymbol{x}_t) \end{pmatrix}.$$

The next stage is to seek an approach to form the analysis errors in terms of the background state combined with a weighting of the observations. Thus for the mixed distribution approach this should be of the form

$$\begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{bl} \end{pmatrix} = \boldsymbol{K}_{mx} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{h}_G(\boldsymbol{x}_a) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{h}_l(\boldsymbol{x}_a) \end{pmatrix}. \tag{21.156}$$

As with the lognormal derivation, we introduce the additive and geometric tangent linear approximations to the Gaussian and lognormal observation operators respectively, into the Jacobian of the mixed distribution cost function and set to zero. This results in

$$0 = \widetilde{\boldsymbol{W}}_b^T \boldsymbol{P}_{fmx}^{-1} \begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{bl} \end{pmatrix} - \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \left(\begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{h}_G(\boldsymbol{x}_b) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{h}_l(\boldsymbol{x}_b) \end{pmatrix} - \boldsymbol{H} \widetilde{\boldsymbol{W}}_o^{-1} \widetilde{\boldsymbol{W}}_b \begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{bl} \end{pmatrix} \right). \tag{21.157}$$

Factorizing (21.157) results in

$$\left[\boldsymbol{P}_{fmx}^{-1} + \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H} \widetilde{\boldsymbol{W}}_b \right] \begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{bl} \end{pmatrix} = \widetilde{\boldsymbol{W}_b^T} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{h}_G(\boldsymbol{x}_b) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{H}_l^T(\boldsymbol{x}_b) \end{pmatrix}. \tag{21.158}$$

Therefore, the analysis errors for the mixed distribution approach are given by

$$\begin{pmatrix} \boldsymbol{x}_{aG} - \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{al} - \ln \boldsymbol{x}_{bl} \end{pmatrix} = \left[\boldsymbol{P}_{fmx}^{-1} + \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b \right]^{-1} \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{h}_G(\boldsymbol{x}_b) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{h}_l(\boldsymbol{x}_b) \end{pmatrix}. \tag{21.159}$$

Thus the Kalman gain matrix for the mixed lognormal-Gaussian approach is

$$\boldsymbol{K}_{mx} \equiv \left[\boldsymbol{P}_{fmx}^{-1} + \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b \right]^{-1} \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1}. \tag{21.160}$$

Through applying the Sherman-Morrison-Woodbury formula it is possible to write (21.160) in the more usable form

$$\boldsymbol{K}_{mx} \equiv \boldsymbol{P}_{fmx} \widetilde{\boldsymbol{W}_b^T} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \left[\widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b \boldsymbol{P}_{fmx} \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} + \boldsymbol{R}_{mx} \right]^{-1}. \tag{21.161}$$

As with the lognormal approach, we shall introduce some notation to simplify the appearance of the derivation for the analysis error covariance matrix. We shall denote $\widetilde{\boldsymbol{H}^T} \equiv \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b$. Using the definition of the analysis errors for the mixed distribution from (21.151), the different forms of tangent

linear approximations presented earlier, as well as the standard version from the Gaussian formulation, results in the mixed distribution analysis error matrix of the form

$$\begin{pmatrix} \boldsymbol{\varepsilon}_{aG} \\ \ln \boldsymbol{\varepsilon}_{al} \end{pmatrix} = \left[\boldsymbol{I} - \boldsymbol{K}_{mx} \widetilde{\boldsymbol{H}^T} \right] \begin{pmatrix} \boldsymbol{\varepsilon}_{bG} \\ \ln \boldsymbol{\varepsilon}_{bl} \end{pmatrix} + \boldsymbol{K}_{mx} \begin{pmatrix} \boldsymbol{\varepsilon}_{oG} \\ \ln \boldsymbol{\varepsilon}_{ol} \end{pmatrix}. \tag{21.162}$$

Thus forming the product of the analysis error vector with its transpose and taking the expectation results in

$$\boldsymbol{P}_{amx} = \left[\boldsymbol{I} - \boldsymbol{K}_{mx} \widetilde{\boldsymbol{H}^T} \right] \boldsymbol{P}_{fmx} \left[\boldsymbol{I} - \boldsymbol{K}_{mx} \widetilde{\boldsymbol{H}^T} \right]^T + \boldsymbol{K}_{mx} \boldsymbol{R}_{mx} \boldsymbol{K}_{mx}^T, \tag{21.163}$$

where following the same arguments for the Gaussian and lognormal case results in the analysis error covariance matrix being of the form

$$\boldsymbol{P}_{amx} = \left[\boldsymbol{I} - \boldsymbol{K}_{mx} \widetilde{\boldsymbol{H}^T} \right] \boldsymbol{P}_{fmx}. \tag{21.164}$$

The final step is to confirm that the analysis error covariance matrix is equivalent to the Hessian of (21.153) in (21.155) which can easily be shown as the expression above is the same in appearance as the standard Gaussian and the lognormal version from the last chapter. Therefore, the analysis error covariance matrix for the mixed distribution approach is given by

$$\boldsymbol{P}_{amx} = \boldsymbol{P}_{fmx}^{-1} + \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \boldsymbol{R}_{mx}^{-1} \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b. \tag{21.165}$$

Thus in summary, the mixed Gaussian-lognormal based Kalman filter equations are given by

$$\boldsymbol{P}_{fmx} = \begin{pmatrix} \boldsymbol{M}\boldsymbol{\varepsilon}_{aG}^{n-1} \\ \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al}^{n-1} \end{pmatrix} \begin{pmatrix} \boldsymbol{M}\boldsymbol{\varepsilon}_{aG}^{n-1} \\ \ln \boldsymbol{M}\boldsymbol{\varepsilon}_{al}^{n-1} \end{pmatrix}^T + \boldsymbol{Q}_{mx}, \tag{21.166}$$

$$\begin{pmatrix} \boldsymbol{x}_{aG} \\ \ln \boldsymbol{x}_{al} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_{bG} \\ \ln \boldsymbol{x}_{bl} \end{pmatrix} + \boldsymbol{K}_{mx} \begin{pmatrix} \boldsymbol{y}_G - \boldsymbol{h}_G(\boldsymbol{x}_b) \\ \ln \boldsymbol{y}_l - \ln \boldsymbol{h}_l(\boldsymbol{x}_b) \end{pmatrix}, \tag{21.167}$$

$$\boldsymbol{K}_{mx} = \boldsymbol{P}_{fmx} \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} \left[\widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b \boldsymbol{P}_{fmx} \widetilde{\boldsymbol{W}}_b^T \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_o^{-T} + \boldsymbol{R}_{mx} \right]^{-1}, \tag{21.168}$$

$$\boldsymbol{P}_{amx} = \left[\boldsymbol{I} - \boldsymbol{K}_{mx} \widetilde{\boldsymbol{W}}_o^{-1} \boldsymbol{H}^T \widetilde{\boldsymbol{W}}_b \right] \boldsymbol{P}_{fmx}. \tag{21.169}$$

In this section it has been shown that it is possible to derive a nonlinear version of the Kalman filter equations to be used with lognormal random variables, as well as with a combination of lognormal and Gaussian random variables. The appearance of the set of equations are similar to the Gaussian form but that the evolution of the analysis error covariance matrix is exact and not through the application of the linearized model. We know consider other approaches for dealing with non-Gaussianity in the form of Gaussian anamorphosis.

21.12 Gaussian Anamorphosis

The work that we present here is from [5] which looks at using Gaussian anamorphosis (GA) in the analysis step of the EnKF. But first off what is a Gaussian anamorphosis? From [5] it is stated that a

GA involves transforming the state variable and observation $\{\mathbf{x}, \mathbf{y}\}$ into new variables $\{\widetilde{\mathbf{x}}, \widetilde{\mathbf{y}}\}$ that present Gaussian features. The EnKF or KF analysis equations are computed using the new variable and the resulting analysis is mapped back into the original space using the inverse of the transformation. We have seen a GA already in this chapter, but it was a crude version as it did not preserve the higher order moment from the original distribution, and that was of coarse the logarithmic transform of a lognormal random variable.

In [5] they present three conditions that ensure optimality of the EnKF/KF: 1) Gaussianity in the prior, 2) linearity of the observation operator, and 3) Gaussianity in the additive observational error density. They start by first assuming that condition 2 and 3 have been met, and so we are focusing on non-Gaussian prior distributions. We have a copy of Figure 1 from [5] in Fig. 21.13 that shows two situations that are challenging for the EnKF analysis step. The likelihood has been kept Gaussian and is centered at the observation.

As described in [5] in the left panel, the prior (blue line) is bimodal, a mixture of two Gaussians centered at $x = -2$ and $x = 2$ with equal variance $\sigma^2 = \frac{1}{4}$. The prior mean is $x = 0$, corresponding to a region where $p_x(x)$ is close to zero. By assimilating an observation (red line) at $y = x = \frac{1}{3}$ the EnKF incorrectly constructs a unimodal analysis PDF (green line) that does not resemble at all the Bayesian posterior (magenta line), where the analysis PDF is centered in a region that the posterior PDF is close to zero.

In the right panel, [5] have a prior that is an exponential distribution with $\lambda = 1$. This is a positive definite variable, and the Bayesian posterior (corresponding to an observation at $y = x = \frac{2}{3}$) correctly captures this information, since $p_{x|y}(x|y) = 0 \; \forall x < 0$. The analysis PDF given by the KF, however, yields non-zero probabilities for negative values of x. In reality, physical observations of a non-negative variable will not be negative. An additive error with Gaussian distribution cannot be used in practice: either a truncated non-symmetric distribution is likely to be used, or negative values will be mapped to zero.

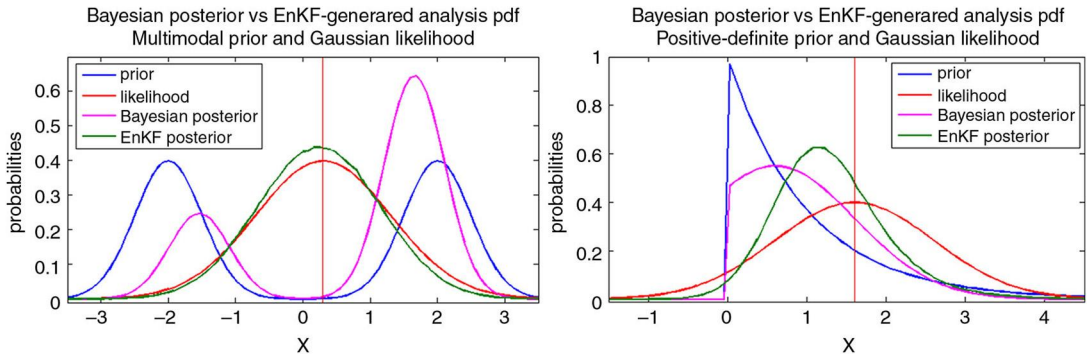


FIGURE 21.13

Copy of Figure 1 from J. Amezcua and P. J. Van Leeuwen, 2014: Gaussian anamorphosis in the analysis step of the EnKF: a joint state-variable/observation approach, *Tellus A*, **66**, 23492. https://www.tandfonline.com/doi/full/10.3402/tellusa.v66.23493. https://creativecommons.org/licenses/by/4.0/.

Thus the GA approach can make use of the integral probability transform theorem (IPT) and solve for the new variable as

$$\widetilde{x} = g(x) = P_x^{-1}(P_x(x)), \tag{21.170}$$

see Appendix A of [5] for the proof of this equation. Therefore, the moments of the target Gaussian variable $\widetilde{x}$ are set to be those of the original ensemble.

Given that it is likely the case that the actual prior distribution, $p_x(x)$ and consequently the CDF, $P_x(x)$, are not known perfectly, then in order to be able to apply the IPT, the first step is to empirically estimate $P_x(x)$ through the ensemble. From this a set of percentiles of the empirical CDF are mapped to the same percentiles CDF of the target distribution. This is possible due to the property of the preservation of percentiles between PDFs (CDFs). The quality of the estimation of $P_x(x)$.

For the multivariate things could become quite difficult, and so one alternative suggested in [5] is to apply the GA univariately so that

$$\widetilde{\mathbf{x}} = g(\mathbf{x}); \quad \begin{pmatrix} \widetilde{x}_1 \\ \widetilde{x}_2 \\ \vdots \\ \widetilde{x}_N \end{pmatrix} = \begin{pmatrix} g_1(x_1) \\ g_2(x_2) \\ \vdots \\ g_N(x_N) \end{pmatrix}. \tag{21.171}$$

It may also be the case that the observations also have to be transformed, and so the GA approach would become:

$$\begin{aligned} \widetilde{\mathbf{x}} &= g_{model}(\mathbf{x}) \\ \widetilde{\mathbf{y}} &= g_{obs}(\mathbf{y}), \end{aligned}$$

where in the transformed space $\widetilde{y}$ and $\widetilde{x}$ are related by the observation operator through the following composition of functions

$$\widetilde{\mathbf{h}} = g_{obs} \circ h \circ g_{model}.$$

In this space for each one of the transformed ensemble members, the EnKF analysis value can be obtained as

$$\widetilde{\mathbf{x}}_m^a = \widetilde{\mathbf{x}}_b + \widetilde{\mathbf{K}}\left(\widetilde{\mathbf{y}}_m - \widetilde{\mathbf{h}}\left(\widetilde{\mathbf{x}}_m^b\right)\right). \tag{21.172}$$

We recommend the reader to [5] for the full discussion about how to use the ensemble to approximate the means in the equation above. We now consider how to choose the anamorphosis functions. What is of concern in the multivariate formulation is the effects these transforms have on the joint characteristics of the state and observations. The motivation in [5] is then is there a transformation that produces a Gaussian posterior PDF $p_{\widetilde{\mathbf{x}}|\widetilde{\mathbf{y}}}(\widetilde{\mathbf{x}}|\widetilde{\mathbf{y}})$ in the transformed space?

For the univariate case $(x, y, \widetilde{x}, \widetilde{y})$ and consider joint bivariate forward transformations of the form

$$\begin{aligned} \widetilde{x} &= g_1(x, y) \\ \widetilde{y} &= g_2(x, y) \end{aligned}, \tag{21.173}$$

with the respective backward transformations:

$$\begin{aligned} x &= q_1(\widetilde{x}, \widetilde{y}) \\ y &= q_2(\widetilde{x}, \widetilde{y}) \end{aligned} . \tag{21.174}$$

Thus if the joint PDF of $\{x, y\}$ in the original space is $p_{xy}(x, y) = p_{t|x}(y|x)\, p_x(x)$, the joint PDF in the transformed space is, again see [5] for the proof:

$$P_{\widetilde{x},\widetilde{y}}(\widetilde{x}, \widetilde{y}) = p_{y|x}(q_2(\widetilde{x}, \widetilde{y}) \mid q_1(\widetilde{x}, \widetilde{y}))\, p_x(q_1(\widetilde{x}, \widetilde{y})) \left| \frac{\partial q_1}{\partial \widetilde{x}} \frac{\partial q_2}{\partial \widetilde{y}} - \frac{\partial q_1}{\partial \widetilde{y}} \frac{\partial q_2}{\partial \widetilde{x}} \right| . \tag{21.175}$$

In [5] they go on to consider different configurations for both the univariate and multivariate situations, and the reader is referred to this paper for more details.

21.13 Gamma-Inverse-Gamma-Gaussian (GIGG) Filter

The Gamma-Inverse-Gamma or GIG as it is more commonly referred to, filter was introduced in [37], which was just after the first edition of this textbook, but it a welcomed addition to this edition. The motivation for this work was the same as ours to introduce the lognormal approach for the variational formulation, and that is to better model the behavior of positive-definite and positive semi-definite random variables. In Figure 1 from [37] has the plots of the distributions of a Gaussian, gamma, inverse-gamma, and the lognormal distributions, and shows that when the mean of the distributions are close to zero with a variance of 1, the Gaussian assigns probabilities to negative values, whereas the other three distributions do not. We have a copy of this figure in Fig. 21.14.

We start by denoting y_{ij}^f as the i-th member of a prior ensemble forecast of the j-th observation y_j^0 of the true state y_j, and it assumed that the continuous PDF of true values y_j that y_{ij}^f randomly samples is a gamma PDF of the form

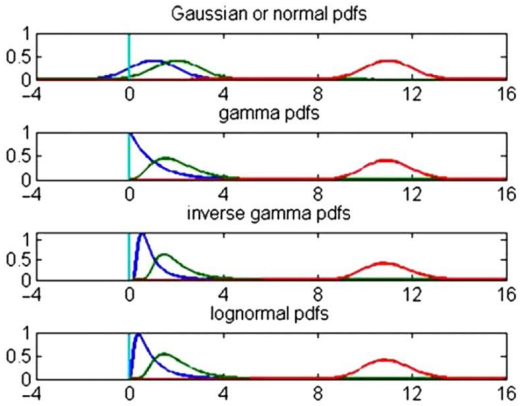


FIGURE 21.14

Copy of figure 1 from [37].

$$\rho_{prior}\left(y_j\right) = \frac{1}{\Gamma\left[\left(P_j^r\right)^{-1}\right]}\frac{1}{y_j}\left(\left(P_j^r\right)^{-1}\frac{y_j}{\mathbb{E}\left[y_j^f\right]}\right)^{\left(P_j^r\right)^{-1}} \exp\left\{-\left(P_j^r\right)^{-1}\frac{y_j}{\mathbb{E}\left[y_j^f\right]}\right\},$$

$$= \frac{1}{\Gamma\left[\left(P_j^r\right)^{-1}\right]}\left(\frac{\left(P_j^r\right)^{-1}}{\mathbb{E}\left[y_j^f\right]}\right)^{\left(P_j^r\right)^{-1}}\left(\left(y_j\right)^{\left(P_j^r\right)^{-1}-1}\exp\left\{-\left(P_j^r\right)^{-1}\frac{y_j}{\mathbb{E}\left[y_j^f\right]}\right\}\right), \tag{21.176}$$

where $\mathbb{E}\left[y_j^f\right]$ is the mean of the prior distribution and $P_j^r = \dfrac{\text{var}\left[y_j\right]}{\mathbb{E}\left[y_j^f\right]^2}$ is referred to as a type 1 relative error variance of the prior distribution. It is then stated that if we have a K-member ensemble forecast that randomly samples the prior distribution then it is possible to approximate these statistics through a sample approach:

$$\mathbb{E}\left[y_j^f\right] \approx \frac{1}{K}\sum_{i=1}^{K} y_{ij}^f \equiv \overline{y_j^f}, \qquad P_j^r \approx \frac{\dfrac{1}{K-1}\displaystyle\sum_{i=1}^{K}\left(y_{ij}^f - \overline{y_j^f}\right)^2}{\left(\overline{y_j^f}\right)^2}.$$

As we saw earlier, the gamma distribution is usually defined by its shape and scale parameters k and θ, where in [37] they use $k \equiv \left(P_j^r\right)^{-1}$ and $\theta \equiv \mathbb{E}\left[y_j^f\right]P_j^r$.

The next step is to assume that the PDF $L\left(y_j^o \middle| y_j\right)$ of observed concentrations of y_j^o given a true y_j is an inverse-gamma distribution so that for a fixed truth y_j

$$L\left(y_j^o \middle| y_j\right) = \left(y_j\left(\widetilde{R}_j^r\right)^{-1}\right)^{\left(\left(\widetilde{R}_j^r\right)^{-1}-1\right)}\frac{\left(y_j^o\right)^{\left(\widetilde{R}_j^r\right)^{-1}-2}\exp\left\{-\left(\widetilde{R}_j^r\right)^{-1}\dfrac{y_j}{y_j^o}\right\}}{\Gamma\left[\left(\widetilde{R}_j^r\right)^{-1}+1\right]},$$

$$= \frac{\left(\widetilde{R}_j^r\right)^{-\left(\left(\widetilde{R}_j^r\right)^{-1}+1\right)}\left(y_j^o\right)^{-\left(\widetilde{R}_j^r\right)^{-1}-2}}{\Gamma\left[\left(\widetilde{R}_j^r\right)^{-1}+1\right]}y_j^{\left(\left(\widetilde{R}_j^r\right)^{-1}+1\right)}\exp\left\{-\left(\widetilde{R}_j^r\right)^{-1}\frac{y_j}{y_j^o}\right\}, \tag{21.177}$$

where $\widetilde{R}_j^r = \dfrac{\text{var}\left[y_j^o - y_j\right]}{\left(y_j\right)^2 + \text{var}\left[y_j^o - y_j\right]} = \dfrac{\text{var}\left[\varepsilon^o\right]}{\left(y_j\right)^2 + \text{var}\left[\varepsilon^o\right]}$ is the type 2 relative observation error variance. When we presented the inverse-gamma earlier it was in terms of the parameters α and β where for this configurations these would be equivalent to $\alpha \equiv \left(\widetilde{R}_j^r\right)^{-1} + 1$ and $\beta \equiv y_j\left(\widetilde{R}_j^r\right)^{-1}$.

Therefore, by Bayes Theorem, the posterior PDF of y_j, given y_j^o, is

$$\rho_{post}\left(y_j \left| y_j^o\right.\right) = \frac{L\left(y_j^o \left| y_j\right.\right)\rho_{prior}\left(y_j\right)}{\int_0^\infty L\left(y_j^o \left| y_j\right.\right)\rho_{prior}\left(y_j\right)dy_j}. \tag{21.178}$$

It is shown in Appendix A of [37] that the posterior PDF is a gamma PDF given by

$$\rho_{post}\left(y_j \left| y_j^o\right.\right) = \frac{1}{\Gamma\left[\left(\Pi_j^r\right)^{-1}\right]}\left(\frac{\left(\widetilde{\Pi}_j^r\right)^{-1}}{\mathbb{E}\left[y_j^a\right]}\right)^{\left(\widetilde{\Pi}_j^r\right)^{-1}}\left(\left(y_j\right)^{\left(\left(\widetilde{\Pi}_j^r\right)^{-1}-1\right)}\exp\left\{-\left(\widetilde{\Pi}_j^r\right)^{-1}\frac{y_j}{\mathbb{E}\left[y_j^a\right]}\right\}\right), \tag{21.179}$$

with type 1 relative error variance $\widetilde{\Pi}_j^r$ and mean $\mathbb{E}\left[y_j^a\right]$ respectively given by

$$\widetilde{\Pi}_j^r = \left(\left(\widetilde{R}_j^r\right)^{-1} + \left(\widetilde{P}_j^r\right)^{-1}\right)^{-1} \equiv \widetilde{P}_j^r - \widetilde{P}_j^r\left(\widetilde{P}_j^r + \widetilde{R}_j^r\right)^{-1}\widetilde{P}_j^r, \tag{21.180}$$

$$\frac{1}{\mathbb{E}\left[y_j^a\right]} = \frac{1}{\mathbb{E}\left[y_j^f\right]} + \frac{\widetilde{P}_j^r}{\widetilde{R}_j^r + \widetilde{P}_j^r}\left(\frac{1}{y_j^o} - \left(\widetilde{R}_j^r + 1\right)\frac{1}{\mathbb{E}\left[y_j^f\right]}\right). \tag{21.181}$$

The next stage in implementing the GIG filter involves the generation of the ensemble for the GIG formulation, where the analysis ensemble whose density is consistent with (21.179) from a forecast ensemble that is drawn from a gamma distribution, it is stated in [37] that posterior sample mean $\overline{y_j^a}$ using

$$\frac{1}{\overline{y_j^a}} - \frac{1}{\overline{y_j^f}} + \frac{\widetilde{P}_j^r}{\widetilde{R}_j^r + \widetilde{P}_j^r}\left(\frac{1}{y_j^o} - \left(\widetilde{R}_j^r + 1\right)\frac{1}{\overline{y_j^f}}\right), \tag{21.182}$$

where the overbars represent the ensemble sample mean. The update of the ensemble perturbation is not so straightforward, and we refer the reader to [37] for the exact details.

The next configuration that is considered in [37] is the inverse-gamma prior with a gamma observation likelihood. This two distributions are given by

$$\rho_{prior} = \frac{\left(\left(\widetilde{P}_j^r\right)^{-1}\mathbb{E}\left[y_j^f\right]\right)^{\left(\left(\widetilde{P}_j^r\right)^{-1}+1\right)}}{\Gamma\left[\left(\widetilde{P}_j^r\right)^{-1}+1\right]}y^{-\left(\left(\widetilde{P}_j^r\right)^{-1}+2\right)}\exp\left\{0\left(\widetilde{P}_j^r\right)^{-1}\frac{\mathbb{E}\left[y_j^f\right]}{y_j}\right\}, \tag{21.183}$$

$$L\left(y_j^p \left| y_j\right.\right) = \left(\frac{1}{\Gamma\left[\left(\widetilde{R}_j^r\right)^{-1}\right]}\left(\widetilde{R}_j^r\right)^{-\left(\widetilde{R}_j^r\right)^{-1}}\left(y_j^o\right)^{\left(\left(\widetilde{R}_j^r\right)^{-1}-1\right)}y^{-\left(\widetilde{R}_j^r\right)^{-1}}\exp\left\{-\left(\widetilde{R}_j^r\right)^{-1}\frac{y_j^o}{y_j}\right\}\right). \tag{21.184}$$

As with GIG it is possible to show, see [37] for the exact details, that the posterior distribution for this set up is given by

$$\rho_{post}\left(y_j \mid y_j^o\right) = \frac{\left(\left(\widetilde{\Pi}_j^r\right)^{-1}\mathbb{E}\left[y_j^a\right]\right)^{\left(\left(\widetilde{\Pi}_j^r\right)^{-1}+1\right)}}{\Gamma\left[\left(\widetilde{\Pi}_j^r\right)^{-1}+1\right]}\left(y_j\right)^{\left(\left(\widetilde{\Pi}_j^r\right)^{-1}+2\right)}\exp\left\{-\left(\widetilde{\Pi}_j^r\right)^{-1}\frac{\mathbb{E}\left[y_j^a\right]}{y_j}\right\}, \tag{21.185}$$

where the posterior type 2 relative error variance $\widetilde{\Pi}_j^r$ is given by

$$\widetilde{\Pi}_j^r = \widetilde{P}_j^r - \widetilde{P}_j^r\left(\widetilde{P}_j^r - R_j^r\right)\widetilde{P}_j^r, \tag{21.186}$$

and

$$\mathbb{E}\left[y_j^a\right] = \mathbb{E}\left[y_j^f\right] + \frac{\widetilde{P}_j^r}{\widetilde{P}_j^r + R_j^r}\left(y_j^o - \mathbb{E}\left[y_j^f\right]\right). \tag{21.187}$$

As with the GIG formulation there is a discussion about the generation of the ensemble to be consistent with this distribution, as well as a presentation of the theory to make this approach multivariate, and the reader is referred to [37] for those details. Our goal here was to introduce another approach to deal with non-Gaussian errors. There are also details about using a Gaussian in conjunction with the GIG formulation to arrive at the GIGG filter.

21.14 Regions of Optimality for Lognormal Descriptive Statistics

As stated earlier, the three descriptive statistics of a left-skewed, or symmetric, distribution satisfy the inequality

$$\mathbf{x}_{mode} \leq \mathbf{x}_{median} \leq \mathbf{x}_{mean}.$$

Therefore, the Gaussian distribution being symmetric implies that the three descriptive statistics are the same. As such we can define an analysis system to find any one of them and they should be equivalent. However, as we have stated and shown, for the lognormal distribution the three statistics are only equivalent when the variance is equal to zero, which would be of no use. We have stated that when the variance is quite small, the median and the mode are quite similar, and that a Gaussian distribution can approximate this form of the lognormal distribution.

As the variance starts to increase, the three statistics start to spread apart and as such it becomes important which statistic to use to minimize the lognormally distributed error. However, as mentioned at the end of the last section, when we tried to implement the mixed full field approach into a 1D VAR retrieval system [221], where we were performing proof of concept through synthetic brightness temperatures that had been generated from a known state with a lognormal error, then given our choice for the a priori state (background state in data assimilation), which was the true state, then the median approach beat the modal approach no matter what size of error variance we used to perturb the true state.

Upon further analysis it became clear that an assumption made in Gaussian-based data assimilation theory/systems **did not** apply to lognormal-based VAR. In Gaussian-based data assimilation systems we assume that our background state is a good approximation to the true state, and because of this,

our first guess for the increment in this VAR system is assumed to be zero, which is implicitly saying that the background state is equal to the true state. This same concept was assumed to carry over to the lognormal formulation, but it soon became apparent that it does not.

In [221] the authors performed a set of experiments with a known truth so that the error in the system could be quantified. They ran experiments with a best Gaussian fit, the transform (median), and the modal system. While the two lognormal-based systems beat the Gaussian, the median approach was always better than the mode. Obviously this was a worrying feature given all the development of the modal lognormal approach. However, it became apparent that the assumption about the a priori/background state was incorrect for lognormal-based data assimilation, but also that it was incorrectly interrupted in the Gaussian formulation.

Upon coding a simple one variable equivalent of the 1D VAR lognormal modal scheme, and matching it with the correct Jacobian and Hessian to use with the Newton-Raphson solver, it became clear that if the a priori state is close to the true state, then the median was always the best statistic to minimize the errors with respect to; however, if the a priori state was larger than the true state, then after a specific value the modal approach was always optimal in minimizing the error. It was also found that if the a priori state was smaller than a specified value, which was a function of the variance, then a cost function based on the **lognormal mean** was optimal in minimizing the error.

What do these findings mean?

i. The a priori state **is not** the best estimate of the true state.
ii. The a priori state is actually the best estimate of the **mean**, $\boldsymbol{\mu}$, of the distribution.
iii. Given the estimates of the error variances, covariances, then there are regions around the a priori state estimate where each of the three descriptive statistics for minimizing lognormal errors are optimal.

In the Gaussian formulation the three descriptive statistics are the same, and so the features listed above did not materialize; however, when skewness starts to play a part in the error distribution, the choice of descriptive statistic, and a priori, state become important.

To help illustrate that there are regions where each of the three descriptive statistics is optimal, we consider a univariate situation. Thus the three cost functions for the mode, median, and mean as

$$J_{mo}(x) = \frac{1}{2}\frac{\left(\ln x - \ln x_{ap}\right)^2}{\sigma_b^2} + \left(\ln x - \ln x_{ap}\right) + \frac{1}{2}\frac{(y - h(x))^2}{\sigma_o^2}, \tag{21.188a}$$

$$J_{md}(x) = \frac{1}{2}\frac{\left(\ln x - \ln x_{ap}\right)^2}{\sigma_b^2} + \frac{1}{2}\frac{(y - h(x))^2}{\sigma_o^2}, \tag{21.188b}$$

$$J_{me}(x) = \frac{1}{2}\frac{\left(\ln x - \ln x_{ap}\right)^2}{\sigma_b^2} - \frac{1}{2}\left(\ln x - \ln x_{ap}\right) + \frac{1}{2}\frac{(y - h(x))^2}{\sigma_o^2}, \tag{21.188c}$$

respectively.

However, the minima of (21.188a)–(21.188c) are found by setting their respective gradients to zero, which are

$$\frac{J_{mo}}{dx} = \left(\frac{1}{x}\right)\left(\frac{\left(\ln x - \ln x_{ap}\right)}{\sigma_b^2} + 1\right) - H\frac{(y - h(x))}{\sigma_o^2}, \tag{21.189a}$$

$$\frac{J_{md}}{dx} = \left(\frac{1}{x}\right)\frac{(\ln x - \ln x_{ap})}{\sigma_b^2} - H\frac{(y - h(x))}{\sigma_o^2}, \tag{21.189b}$$

$$\frac{J_{me}}{dx} = \left(\frac{1}{x}\right)\left(\frac{(\ln x - \ln x_{ap})}{\sigma_b^2} - \frac{1}{2}\right) - H\frac{(y - h(x))}{\sigma_o^2}, \tag{21.189c}$$

respectively.

There are three free parameters in (21.189a)–(21.189c) such that the minimum of the cost function is at the *true* state; the a priori state, the background error variance, and the observational error variance. Thus, given the incorrect estimates for two of the parameters, it is possible to set the remaining parameter's value such that it compensates for the incorrect values for the other parameters such that the minimum of the cost function will be at the true state.

If we consider the case where of no observational error (perfect observations), then at the true solution, x_t, the observational component is zero. Therefore, there are two remaining parameters that are free to be chosen so that the minimums of (21.188a)–(21.188c) are at x_t.

The first parameter that we set to compensate for the other remaining parameter is the a priori state, x_{ap}. This means that bounds are sought so that it is clear which of the three statistics is the best option to use for these values. As the mean is $\exp\left\{\ln x_{ap} + \frac{\sigma_b^2}{2}\right\} \equiv x_{ap}\exp\left\{\frac{\sigma_b^2}{2}\right\}$ then the value of x_{ap} such that the zero of (21.189c) is at x_t is $x_{ap} = x_t \exp\left\{-\frac{\sigma_b^2}{2}\right\}$.

For the median approach, if $x_{ap} = x_t$ then the minimum of (21.189b) will occur at x_t. For the modal approach, if $x_{ap} = x_t \exp\{\sigma_b^2\}$ then the minimum of (21.189a) is at x_t.

Given these three values, we now investigate what the implications are. The reason for presenting the solutions in this order is to show the areas that the three parameters are the best approach; starting from near zero with the mean, transitioning into the median, and finally into the mode. Looking at the definitions of the three statistics for the lognormal distribution, it becomes clear that the mean will be the best statistic for values of x_{ap} greater than zero to $x_{ap} = x_t \exp\left\{-\frac{\sigma_b^2}{4}\right\}$. At this point the error associated with the median and the mean are equal in magnitude but opposite in sign. The median now takes over to the halfway point between the definition of the median and the mode. This point is $x_{ap} = x_t \exp\left\{\frac{\sigma_b^2}{2}\right\}$. There is also a point where the mode takes over having a smaller error than the mean but still larger than the median. This occurs at $x_{ap} = x_t \exp\left\{\frac{\sigma_b^2}{4}\right\}$.

To illustrate the ranges for which the three different statistics minimize the error the best, the errors associated with solving (21.189a)–(21.189c) using the Newton-Raphson method with the nonlinear observations operator $h(x) = x^2$ are presented in Fig. 21.15. All first guesses are $x_{fg} = 1$, with $\sigma_b^2 = 2$ and $\sigma_o^2 = 0.01$. The true state is generated from a random lognormal distribution random number generator in MATLAB using $\mu = -0.75$ and $\sigma = 1.4454$. These values have been found from real data associated with C1DOE [221]. The true solution is $x_t = 6.5$.

In Fig. 21.15A we have plotted the errors associated with different values of x_{ap}, where we are stepping the a priori state as $x_{ap} = x_t \exp\{(j - 10) * 0.05\sigma_b^2\}$ for $j = 1, 2, \ldots, 100$, for the different values for the minima of (21.188a)–(21.188c) associated with each of these values for the a priori state.

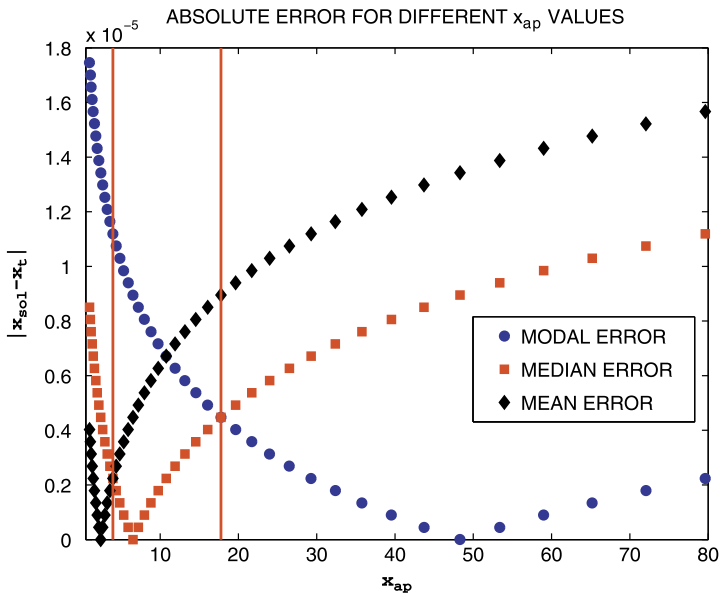


FIGURE 21.15

Plots of the optimal regions for each of the three descriptive statistics for the lognormal based 1D VAR.

In Fig. 21.15B we have plotted the absolute errors. This is to illustrate where the different statistics take over from each other, in magnitude, in minimizing the errors.

A clear feature present in both plots is that the area where the mean is the optimal statistic is quite small. The median is optimal between $x_t \exp\left\{-\frac{\sigma_b^2}{4}\right\}$ and $x_t \exp\left\{\frac{\sigma_b^2}{2}\right\}$, where at the upper bound the mode takes over. These two bounds are plotted in Fig. 21.15B to illustrate the range which each statistic is optimal. Another clear feature is that for this configuration, for the mode to minimize the errors, the a priori state has to be quite large relative to the true state. Therefore, these findings are illustrating that in the lognormal formulation the a priori state is not an approximation to the true state, as is the case for the Gaussian VAR scheme, but is actually an approximation to the **mean of the lognormal distribution of the true state**. This is also true for the Gaussian case, but there the three statistics are the same. Therefore, in the lognormal situation it appears that the median approach is optimal at minimizing the errors if the x_{ap} is very close to the true solution. If the a priori state is much larger than the true state, then the median approach is contributing to the inaccuracy of the solution.

To illustrate how strong the assumption that the a priori state has to be quite close to the true state for the median approach, we have plotted the absolute error plots for five different values for σ_b^2, 0.1, 0.5, 1, 1.5, and 2, in Fig. 21.16. The reason for presenting these plots it to illustrate that the region where the median is optimal can be quite small for smaller values of the estimated background error variance; however, for values of the a priori state that are greater than the upper bound of the median, the modal approach will always have the smallest error.

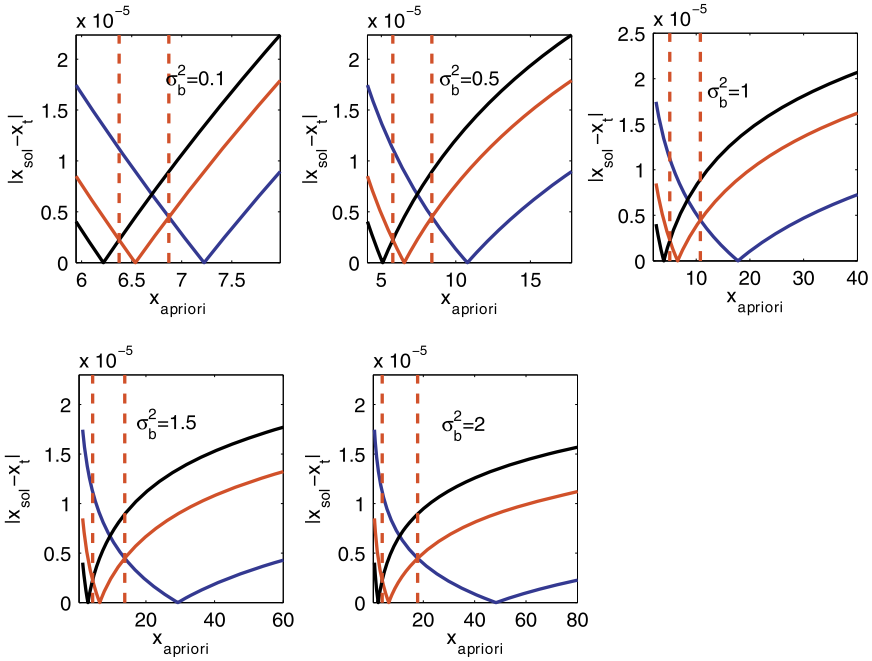


FIGURE 21.16

Plots of the optimal regions for the median as the variance increases.

However, it is possible to bound the background error variance instead of x_{ap}. To obtain these values we consider the Jacobian equations in (21.189a)–(21.189c) evaluated at x_t. The first important feature is that (21.189b) cannot be rearranged so that σ_b^2 can be isolated, implying for the median approach, there is no optimal value for σ_b^2. Therefore, the median approach can only be optimal if $x_{ap} = x_t$. For the modal approach (21.189a), if $\sigma_b^2 = \ln\left(\frac{x_{ap}}{x_t}\right)$, then it is possible to converge for the lognormal approach $\forall x_{ap} > x_t$. Finally for the mean approach, the relationship for σ_b^2 is $\sigma_b^2 = \ln\left[\left(\frac{x_t}{x_{ap}}\right)^2\right]$, $\forall x_t > x_{ap}$.

In [133] it is shown that when trying to solve for the optimized background error variance situation there appears to be a sensitivity to the value of the first guess for the Newton-Raphson solver. This sensitivity is investigated quite substantially in [133], where it is revealed that the sensitivity to the first guess is also present when we optimize for the case where there measurement errors for the observations. The investigation showed that there is a Newton fractal present when the cost function for both lognormal- and Gaussian-based VAR is optimal. This sensitivity to the first guess can result in a halving of the figures of accuracy of the solution.

We now consider the case where we have measurement errors, ε_m, associated with the observations and derive estimates to optimize the three parameters x_{ap}, σ_b^2 and σ_o^2 for the modal approach. Therefore, the cost function for this situation is

$$J_{obsm}(x) = \frac{1}{2}\left(\frac{\left(\ln x - \ln x_{ap}\right)^2}{\sigma_b^2}\right) + \left(\ln x - \ln x_{ap}\right) + \frac{1}{2}\left(\frac{\left(y_t + \varepsilon_m - h(x)\right)^2}{\sigma_o^2}\right). \tag{21.190}$$

As in the case of perfect observations, it is the Jacobian of (21.190) evaluated at $x = x_t$, that enables us to derive the optimal expressions for x_{ap}, σ_b^2 and σ_o^2. Thus the Jacobian of (21.190) is

$$\left.\frac{\partial J_{obsm}}{\partial x}\right|_{x=x_t} = \frac{1}{x_t}\left(\frac{\left(\ln x_t - \ln x_{ap}\right)}{\sigma_b^2} + 1\right) - H|_{x=x_t}\frac{\varepsilon_m}{\sigma_o^2} = 0, \tag{21.191}$$

where $y_t - h(x) = 0$.

Through rearranging the optimal values for x_{ap}, σ_b^2, and σ_o for the modal approach are

$$x_{ap,opt} \equiv x_t \exp\left\{\sigma_b^2 - \frac{x_t\ H|_{x=x_t}\,\varepsilon_m\sigma_b^2}{\sigma_o^2}\right\}, \tag{21.192a}$$

$$\sigma_{b,opt}^2 \equiv \left(1 - \frac{x_t\ H|_{x=x_t}\,\varepsilon_m}{\sigma_o^2}\right)^{-1}\left(\ln x_{ap} - \ln x_t\right), \tag{21.192b}$$

$$\sigma_{o,opt}^2 \equiv \frac{x_t\ H|_{x=x_t}\,\varepsilon_m\sigma_b^2}{\ln x_t - \ln x_{ap} + \sigma_b^2}, \quad \text{for} \quad x_t \neq x_{ap}\exp\left\{-\sigma_b^2\right\}. \tag{21.192c}$$

An important feature to note here is that when (21.192a) and (21.192b) are evaluated at $\varepsilon_m = 0$, the expressions become those identified for the perfect observation case. When (21.192c) is evaluated at $\varepsilon_m = 0$, there is no statistically consistent solution; however, this is again consistent with the results from the perfect observations situation.

It was during the testing of these optimal configurations for the free parameters that the sensitivity to the initial guess became quite noticeable and led to a more in-depth study of the Newton fractal associated with these situations. Note that a similar fractal was also found for the Gaussian formulation of 1D VAR and also for the case of representative error for the lognormal formulation.

While we shall not go into details here about the Newton fractal, we shall leave you with a plot of the error-plane for positive and negative measurement errors against estimates for the a priori state, where the function is the value of the difference between two iterations at convergence of the Newton-Raphson solver from [133].

21.15 Summary

In this chapter we have introduced the theory that enables the variational form of data assimilation to be applicable for lognormally distributed background, observational, and model errors. The lognormal variational theory was combined with the established Gaussian theory to allow for background, observational, and model errors to follow a mixed Gaussian-lognormal distribution. The starting point for these derivations came from the geometric form for lognormal errors in [68].

We showed that the definition of the lognormally distributed observational errors from [68] was extended in [135] to allow for the nonlinear observation operator for a modal approach through the Bayesian derivation of 3D VAR from [259]. Returning to the mixed distribution, which was defined and proved to be a probability density function in [136], we were able to show that it was possible

to define a Bayesian problem that had a mixed Gaussian-lognormal conditional probability density function for the observational errors.

As we were progressing through this theory, we also acknowledged that there was an alternative approach, referred to the logarithmic transform technique, which uses the change of variable of x to $\ln x$, which would be a Gaussian random variable but that upon inversion back to the lognormal space, then the analysis variable was an estimation to the median of the lognormal analysis (posterior) distribution.

All of the techniques described above have been extended to the full field 4D VAR case where it have been shown that it is possible for the modal approach to outperform the median/transform approach when the observational error variance becomes large in the Lorenz 1963 model, but with the background error covariance matrix being updated after each assimilation window.

For the theory to be considered operationally viable for atmospheric and ocean numerical prediction centers, we required a form of incremental 3D and 4D VAR. We have presented two different formulations for a geometric-based increment. The first is consistent with finding the median approach through using the property that the exponential of a lognormal random variable is a Gaussian random variable [409]. The second form of a possible increment with which to form an incremental lognormal 3D and 4D VAR system was a lognormally distributed increment, which then enabled the cost function for the lognormal mode to be consistently incrementalized [132].

However, we showed in the last section that the concept that the background state $\mathbf{x}_b$ should be a good estimate of the true state was not the correct interpretation for this state. We showed in that section that it should be an approximation to the mean of the true state for a non-median based lognormal approach. We showed that if the a priori state is close to the true state, then the median is the best

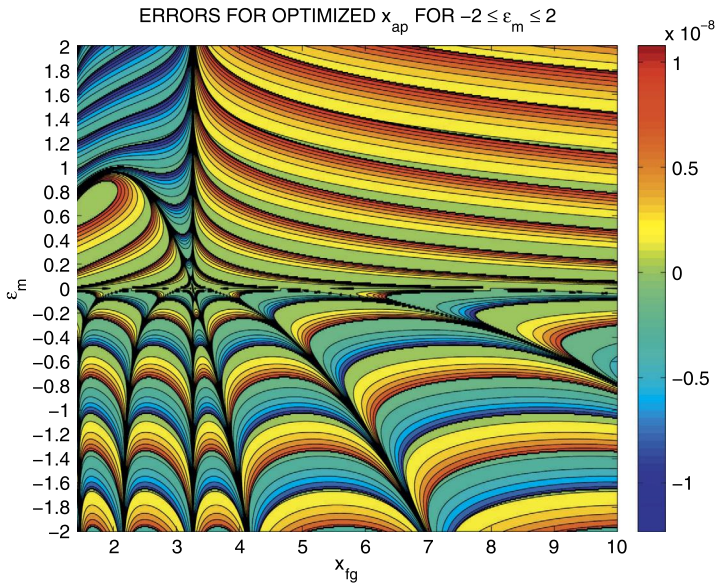


FIGURE 21.17

Plot of the error plane for positive and negative measurement errors to highlight the Newton fractal.

descriptive statistic to minimize the error. If, however, the a priori state is much larger than the true state, then the modal approach is the optimal choice to minimize the errors.

It was during the investigation of the optimal regions for all three descriptive statistics for the lognormal distribution that if a cost function had been optimized—by this it is meant that if the free parameters had been chosen such that the minimum of the cost function was at the true state—then a sensitivity to the first guess to the Newton-Raphson solver becomes noticeable. It is possible to illustrate graphically the regions of optimal convergence for the iterative solver and we have presented one such illustration in Fig. 21.17.

We have introduced the derivation of the lognormal and the mixed Gaussian-lognormal version of the Kalman filter. We have also introduced Gaussian anamorphosis, along with the Gamma-Inverse-Gamma-Gaussian (GIGG) filter as approaches that be applied to non-Gaussian situations.

However, the variational approach is not the only way to deal with non-Gaussian errors in a data assimilation situation. While we have not presented it in this book, it is possible to extend the representer theory from Chapter 18 to allow for both lognormal and mixed Gaussian-lognormal errors, in an alternative incremental form to that shown here. This work has not yet been sent to peer review, but we make the readers aware of this cutting-edge research. However, the representer methods aside, we now consider the theory of Markov chain Monte Carlo and particle filter approaches which do not make any assumptions of which probability density functions are involved; these approaches are introduced in the next chapter.

This page intentionally left blank

CHAPTER

22 Markov Chain Monte Carlo, Particle Filters, Particle Smoothers, and Sigma Point Filters

Contents

Over the last few chapters, we have introduced versions of data assimilation that are either empirical, statistical, variational, or ensemble based. In most of the theory associated with these approaches, we had to make the Gaussianity assumption for the probabilistic behavior of the errors; the exception to this rule came in the last chapter, where we presented variational-based approaches derived by the author with colleagues from CIRA, that allows for lognormal and mixed Gaussian-lognormal errors, as well as the equivalent Kalman filter for this distributions. This was then extended to the MLEF. We also introduced the Gaussian anamorphosis, as well as the GIGG filter.

One of the downside of the variational approach in the last chapter is that it was only for the lognormal, or the mixed lognormal-Gaussian distribution, and deriving a new cost function for other distributions may not be as straightforward as for the lognormal case. For the **Markov chain Monte Carlo (MCMC)**, methods we do not make any assumption of Gaussianity, or lognormality for the probabilistic behavior of the random variables involved. Therefore, these approaches are designed to allow

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00026-7

for the relaxation of the Gaussian assumption. However, there are problems with these approaches from an operational numerical prediction point of view, which we shall explain later, but much progress has been made since the first edition with particle filters towards being operationally viable.

In this edition of the text book we have expanded this chapter to now include the localized particle filter, as well as to present the sigma point Kalman filters, along with the unscented Kalman filter.

22.1 Markov Chain Monte Carlo Methods

The starting point for the MCMC methods is to introduce the concept of white noise and a random walk. The definitions presented here come from [78].

Definition 22.1. A **white noise** process is a discrete time random process $\{W_t : \ldots, -1., 0, 1, \ldots\}$ whose elements are mutually independent with a common probability density function. Note that this distribution does **not** have to be Gaussian.

Definition 22.2. A time series $\{Y_t\}$ is said to be a **random walk** if

$$Y_t = Y_{t-1} + W_t, \quad t = 0, 1, \ldots, \tag{22.1}$$

where W_t is a white noise process with mean μ_w and variance σ_w^2. It is possible to show that $\mathbb{E}[Y_t] = t\mu_w$ and $Var[Y_t] = t\sigma_w^2$. This is to say that both the mean and the variance are a function of time; thus this implies that the process is non-stationary.

As mentioned in Chapter 15, data assimilation can be considered as an inverse problem, as stated by Tarantola and Valette in their 1982 papers [430,431], and it is Tarantola who we turn to now. In his 2005 book [429], Tarantola has a brief chapter on Monte Carlo (MC) methods, and we shall present some of his key points here.

MC methods are quite often used in the numerical evaluation of integrals in large-dimensional spaces. In Appendix 6.9 of [429] there is a good description of the MC integration approach. The basis of the MC approach for integration is to consider a s-dimensional manifold, $\mathcal{Q}$, with coordinates $\left\{q^1, q^2, \ldots, q^s\right\}$. For a point of the manifold that is denoted as $\mathbf{q} \in \mathcal{Q}$, let $\phi(\mathbf{q})$ be an arbitrary scalar function defined over $\mathcal{Q}$, and assume that we require the evaluation of

$$S = \int_{\mathcal{Q}} d\mathbf{q}\phi(\mathbf{q}) = \underbrace{\int dq^1 \cdots \int dq^s}_{\mathcal{Q}} \phi\left(\mathbf{q^1}, \mathbf{q^2}, \ldots, \mathbf{q^s}\right). \tag{22.2}$$

Normally, if the manifold was a finite volume, then we could numerically approximate S through a regular grid of points in $\mathcal{Q}$; we would then compute $\phi(\mathbf{q})$ at each grid point and apply a numerical integration technique. However, as the number of grid points is a rapidly increasing function of the dimensionality of the problem, the numerical approximation becomes impractical.

The MC method of integration consists of replacing the regular grid of points with a pseudo-random grid generated by a pseudo-random number generator.

If we let $p(\mathbf{q})$ be an arbitrary normed probability density over $\mathcal{Q}$, where by normed it is meant that $d\mathbf{q}\mathbf{p}(\mathbf{q}) = \mathbf{1}$, that is used to generate pseudo-random points over $\mathcal{Q}$, along with $\psi(\mathbf{q}) \equiv \frac{\phi(\mathbf{q})}{p(\mathbf{q})}$, then the sum that we wish to evaluate can be written as

$$S = \int_{\mathcal{Q}} d\mathbf{q}\mathbf{p}(\mathbf{q})\psi(\mathbf{q}). \tag{22.3}$$

Now let $\mathbf{q_1}, \mathbf{q_2}, \ldots, \mathbf{q_N}$ be a set of N points that are collectively independent and randomly distributed over $\mathcal{Q}$ with a probability density $p(\mathbf{q})$, with the following three measures [429]:

$$\psi_n \psi(\mathbf{q}), \quad S_N = \frac{1}{N}\sum_{n=1}^{N}\psi_n, \quad V_N = \frac{N}{N+1}\left(\frac{1}{N}\sum_{n=1}^{N}\left(\psi_n^2 - S_n^2\right)\right). \tag{22.4}$$

When we are considering a large number of dimensions, then representing a PDF is impossible; however, we can sample, which we shall explain briefly from the explanation presented in [309].

Let us consider a random process that selects points in the model space. If the probability of selecting point i is p_i, then the points selected by the process are called *samples* of the probability distribution $\{p_i\}$. Depending on the random process, successive samples $i, j, k, \ldots$, may be dependent or independent, in the sense that the probability of sampling k may or may not depend on the fact that i and j have just been sampled.

In the first part of this section, we introduce the mathematical definition of a *random walk*. In [309] it is stated that the process of a random walk defines a graph in the model space. We have presented a random walk schematic similar to the one in [309] in Fig. 22.1. All models in the discrete model space are nodes of the graph in Fig. 22.1, and the edges of the graph define the possible steps of the random walk. Now the graph defines the **neighborhood** of a model as the set of all models directly connected to it. Sampling is then made by defining a random walk on the graph.

We now define the probability P_{ij} for the *random walker* to go to point i if it is currently at point j. The probability P_{ij} is the **transition probability**. We should note that it is possible for the random walker to stay at the same point.

If we now consider a random walk, defined by the transition probabilities $\left\{P_{ij}\right\}$, and assume that the model where the random walk is initiated is only known probabilistically, then there is a probability, q_i, that the random walk is initiated at point i. As the number of steps tends to infinity, the probability that the random walker is at point i will converge to some probability, p_i, where $\{p_i\}$ is an **equilibrium probability distribution** of $\left\{P_{ij}\right\}$.

If, at some point, the probability for the random walker to be at point j is p_j, and the transition probabilities are P_{ij}, and if we define a probability $f_{ij} = P_{ij}p_j$, then this is the probability that the ran-

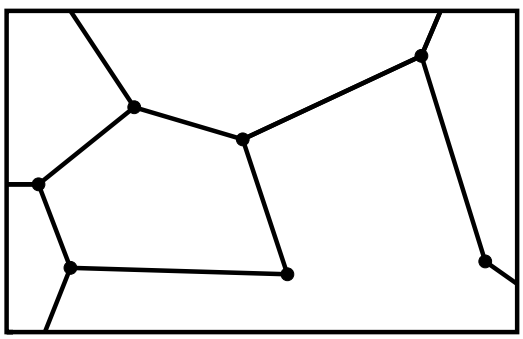


FIGURE 22.1

Illustration of the path that a random walker could take in a general domain.

dom walker will take a walk (transition) from point j to point i; while P_{ij} is the conditional probability that the random walker goes to point i if the walker is at point j, the probability f_{ij} is the unconditional probability that the next step will be a transition to i from j [309].

In [309] a couple of possible walks are presented that the random walker could take, and we briefly summarize them here.

Naïve walk

If we consider the graph in Fig. 22.1, and denote the number of neighbors of point i by n_i—note this includes the point i itself—then a naïve walk is one where a walker who is at point j moves to one of point j's neighbors, say neighbor i, at random where each neighbor has an equal probability of being selected, that is to say chosen uniformly at random. It is proven in [309] that this type of random walk equilibrates at the probability distribution given by $p_i = \frac{n_i}{\sum_j n_j}$, where n_j is the number of neighbors at point j.

Uniform walks

We now have a random walker that, while they are at point j, first chooses, uniformly at random, one of j's neighbors, say i, and then uses the following **rule** to decide if they move to point i or stay at j:

1. If $n_i \leq n_j$, that is to say that if point i has less neighbors than point j, then always move to i.
2. If $n_i > n_j$, that is to say that if point i has more neighbors than point j, then make a random decision to move to point i or stay at point j, with the probability $\frac{n_i}{n_j}$.

Returning to the integration problem, we have the possible difficulty that when sampling large-dimensional spaces then it is easy to underestimate the small region of significant probability. There are two problems with MC sampling if a PDF is in a large dimensional space: (1) locating the regions of significant probability, and (2) sampling all the regions densely enough.

The most difficult problem with the MC approach, according to Tarantola, is discovering the location of the regions of significant probability. However, when we have been able to come close to a region of significant probability, it is possible to perform a random walk, which can be seen as a sort of Brownian motion, that is efficient in exploring the region, and avoid leaving it. There are two well-known algorithms that can be used to stay in the region of interest; these are the **Gibbs sampler** and the **Metropolis-Hastings algorithm**.

22.1.1 MC Methods for Inverse Problems

In data assimilation, we have a situation where we have a probability density $\rho(\mathbf{x})$ that describes the a priori information of the model states or parameters, and a probability density $\rho(\mathbf{y})$ describing the information on the observations. As we have seen many times, the solution to the inverse problem is given by a posterior probability density $\varrho(\mathbf{x})$ that is equal to the normalized product of the a priori probability density and a likelihood function $L(\mathbf{x})$, such that

$$\varrho(\mathbf{x}) = k\rho(\mathbf{x}) L(\mathbf{x}), \tag{22.5}$$

where the likelihood function is a measure of how well the model fits the observations.

It is shown in [429] that the inverse problem can be expressed as

$$\varrho(\mathbf{x}) = k\rho(\mathbf{x}) \int_{\mathcal{S}} d\mathbf{y} \frac{\rho(\mathbf{y})\, \rho(\mathbf{y}|\mathbf{x})}{\tau(\mathbf{y})}, \tag{22.6}$$

where $\tau(\mathbf{y})$ is the homogeneous probability density in the observation manifold.

We now have to consider how to sample the distributions.

22.1.2 Sample Methods

In this subsection we consider four methods that have been used in the geosciences: the rejection method, the sequential realization, the Gibbs sampler, and the Metropolis-Hastings algorithm.

Rejection method

The rejection method starts by generating samples $x_1, x_2, \ldots$ of the homogeneous density $\tau(x)$. Then each sample is submitted to the possibility of rejection, where the probability of the sample x_i as being accepted is equal to

$$P = \frac{\frac{f(x_i)}{\tau(x_i)}}{\left(\frac{f}{\tau}\right)_{max}}, \tag{22.7}$$

where $\left(\frac{f}{\tau}\right)_{max}$ represents the maximum for all the values of $\frac{f(x_i)}{\tau(x_i)}$.

Sequential realization

In this approach we are using the property that a general n-dimensional PDF, $f_n(x_1, x_2, \ldots, x_n)$, can be decomposed into the product of a one-dimensional marginal distribution and a series of one-dimensional conditional PDFs

$$f_n(x_1, x_2, \ldots, x_n) = f_1(x_1)\, f_{1|1} f(x_2|x_1)\, f_{1|2}(x_3|x_1, x_2) \ldots f_{1|n-1}\left(x_n|x_1, x_2, \ldots, x_{n-1}\right). \tag{22.8}$$

This process works by starting with the generation of a one-dimensional sample for the variable x_1 using the marginal distribution $f_1(x_1)$, which gives a value $x_{1,0}$. Using this value, we generate a one-dimensional sample of x_2, using the conditional PDF $f_{1|1}\left(x_2|x_{1,0}\right)$, which gives a value for $x_{2,0}$. Next we generate a sample for the variable x_3 using the conditional PDF $f_{1|2}\left(x_3|x_{1,0}, x_{2,0}\right)$, which gives a value for $x_{3,0}$, and so on until we arrive at the last condition distribution $f_{1|n-1}\left(x_n|x_{1,0}, x_{2,0}, \ldots, x_{n-1,0}\right)$, which results in $x_{n,0}$. In this manner a point $\left\{x_{1,0}, x_{2,0}, \ldots, x_{n-1,0}, x_{n,0}\right\}$ is generated that is a sample of the original PDF, $f_n(x_1, x_2, \ldots, x_n)$.

Gibbs sampler

Let $f\left(x_{1,k}, x_{2,k}, \ldots, x_{n,k}\right)$ be the PDF that we wish to sample and let $\mathbf{x}_k = \left\{x_{1,k}, x_{2,k}, \ldots, x_{n,k}\right\}$ be the last point that we visited. Now define a random line in the space that passes through the current point $\mathbf{x}_k$. Along that line there is a one-dimensional conditional PDF. A sample is then generated along this one-dimensional PDF, which results in a new point $\mathbf{x}_{k+1}$. Tarantola has some comments about the efficiency of this method for higher dimensional problems, and the reader is referred to his book for those comments [429].

Metropolis-Hastings algorithm

This sampling algorithm is used quite extensively in the application of inverse modeling across multiple geosciences disciplines [277,310,329,344,360,426,455,456,481]. The Metropolis-Hastings algorithm, was presented in Metropolis et al. in 1953 [295] and refined in Hastings [167].

The idea behind a MCMC method is to use a Markov Chain when sampling the posterior distribution in the Bayesian problem. This means that the sample point only has information from the previous sample, and not all of the sample before that one. We introduced a couple of simpler random walks earlier in this chapter. The basic idea of a random walk, which can be seen as a sort of Brownian motion, and if the walk is unmodified then we would sample some initial probability. We then use some probabilistic rule to modify the walk, where some proposed moves would be rejected, while some would be accepted, in such a way that the random walker samples the target distribution. One of the most efficient rules for achieving the goal is the Metropolis-Hastings algorithm.

If we consider the case where we have two probability densities $f(\mathbf{x})$ and $g(\mathbf{x})$ together with their homogeneous limit, $\tau(\mathbf{x})$—see Tarantola [429] for the meaning of the homogeneous limits—then the question becomes how should we modify the algorithm in order to obtain samples of the conjunction of the two probability densities given by

$$h(\mathbf{x}) = k\frac{f(\mathbf{x})\, g(\mathbf{x})}{\tau(\mathbf{x})}. \tag{22.9}$$

The criteria used with this algorithm will depend not on the values of the probability density, $g(\mathbf{x})$, but on the values of the associated likelihood function, which is defined as

$$\gamma(\mathbf{x}) = \frac{g(\mathbf{x})}{\tau(\mathbf{x})}. \tag{22.10}$$

It is assumed that some random rules define a random walk that samples the probability density function $f(\mathbf{x})$. At a given step, the random walker is a point $\mathbf{x}_i$, and the application of the rules would lead to a transition to point $\mathbf{x}_j$. When all such proposed transitions of $\mathbf{x}_i \rightarrow \mathbf{x}_j$ are accepted, the random walker will sample the probability density $f(\mathbf{x})$. Now, instead of always accepting the proposed transition $\mathbf{x}_i \rightarrow \mathbf{x}_j$, we could reject the transition by using the following rule, which will allow us to decide if the random walker is allowed to move to $\mathbf{x}_j$ or if they have to stay at $\mathbf{x}_i$.

i If $\gamma(\mathbf{x}_j) \geq \gamma(\mathbf{x}_i)$, then accept the proposed transition to $\mathbf{x}_j$.
ii If $\gamma(\mathbf{x}_j) < \gamma(\mathbf{x})$, then decide randomly to move to $\mathbf{x}_i$, or to stay at $\mathbf{x}_i$ with the following probability of accepting the move to $\mathbf{x}_j$:

$$P_{ij} = \frac{\gamma(\mathbf{x}_i)}{\gamma(\mathbf{x}_j)}. \tag{22.11}$$

If we now return to the Bayesian problem for the posterior distribution, we have

$$\varrho(\mathbf{x}) = k\rho(\mathbf{x})\, L(\mathbf{y}|\mathbf{x}),$$

where we still have that $\rho(\mathbf{x})$ represents the a priori of the model values, and $L(\mathbf{y}\,|\,\mathbf{x})$ is our measure of the goodness of the model values $\mathbf{x}$ is at fitting to our observations $\mathbf{y}$.

The steps that we have to follow to apply the MCMC method to approximate the inverse problem is as follows:

- Sample the a priori distribution, $\rho(\mathbf{x})$, through some simple sampling scheme, or through applying the Gibbs sampler or the Metropolis-Hastings algorithm.
- Given the points from the prior distribution, we now evaluate the likelihood subject to the following rules:

  1. If $L(\mathbf{x}_j) \geq L(\mathbf{x}_i)$, then accept the proposed transitions to $\mathbf{x}_j$.
  2. If $L(\mathbf{x}_j) < L(\mathbf{x}_i)$, then decide randomly to move to $\mathbf{x}_j$ or to stay at $\mathbf{x}_i$, with the probability of accepting the move to $\mathbf{x}_j$ being

$$P_{ij} = \frac{L(\mathbf{x}_j)}{L(\mathbf{x}_i)}.$$

  The ratio above is often known as the **acceptance ratio** [344].

Then the random walker samples the posterior PDF $\varrho(\mathbf{x})$.

An important feature here is that the problem we are solving is for the minimum variance [424]. It is possible to find the maximum likelihood estimate using the MCMC approach, which is referred to as using **simulated annealing** [424,429]. However, we do disagree with a sweeping statement made in Tarantola's book where he says

> It is my point of view that if we are able to sample the probability density $\varrho(\mathbf{x})$ we should not be interested in the maximum likelihood point. As any central estimator (like the mean or the median), the maximum likelihood point is of very little interest when dealing with complex probability distributions.

First, the mean is not a central estimator; this is only true for symmetric distributions, and in fact only the median is the central, unbiased estimator. Second, work with the lognormal distribution has shown that the mean is unbounded and in the tails for large variances. We should recall that in data assimilation we are trying to minimize the posterior errors, and as such in the variational methods we seek the state with the highest probability of doing this.

Another feature to mention about the MCMC methods is the **burn in period**. When starting a MCMC algorithm, ideally the starting value is chosen as well as it can be, in order to avoid time-consuming initial wandering in uninteresting areas of the model space [381,425]. As an aside, there is a very good history of the application of MC and MCMC methods in the geosciences in [381], which also cites the work of Tarantola and Valette in 1982.

Choices for prior distribution and likelihood functions

We could take any distribution type to represent the prior distribution; quite often the easiest one to use is the uniform distribution [344], as this enables the prior distributions in the acceptance ratio. Other choices include the Gaussian distribution, as it is easy to evaluate the probabilities associated with the different points.

With respect to the likelihood function, we have to recall that this function describes the uncertainty associated with the observations, given the value for the model state, or a function of the model state. Therefore, the Gaussian likelihood function is given by

$$L(\mathbf{y} \mid \mathbf{x}) \equiv \frac{1}{2} (\mathbf{y} - \mathbf{h}(\mathbf{x}))^T \mathbf{R}^{-1} (\mathbf{y} - \mathbf{h}(\mathbf{x})). \tag{22.12}$$

Therefore, we say that the observational errors are Gaussian distributed.

For certain applications of the MCMC method, the Gaussian likelihood function may not be the best choice. Posselt et al. [344], use the theory from Fletcher and Zupanski [135] to define a lognormal distribution-based likelihood function given by

$$L(\mathbf{x}) = \ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}) + \frac{1}{2} (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x}))^T \mathbf{R}_L^{-1} (\ln \mathbf{y} - \ln \mathbf{h}(\mathbf{x})), \tag{22.13}$$

to compare with the Gaussian-based likelihood function to explore the error characteristics of thin ice cloud property retrievals. However, we should note that the expression in (22.13) is for the mode of the lognormal distribution, not the mean.

22.1.3 Application of MCMC in the Geosciences

As mentioned in the last section, there is a good history of MCMC methods summarized in Sambridge and Mosegaard [381], where they present results with respect to a seismology application. In [360] they apply the MCMC approach to obtain a mapping of subsurface regions where the electrical resistivity is changing.

Tamminen [424] applies the MCMC method to an ozone retrieval with the Global Ozone Monitoring by Occultation of Stars. The occultation that they refer to is with respect to a bright and a dim star as their light passes through the atmosphere as the satellite descends. Therefore, given the change in the spectrums, it is possible to invert out the amount of ozone in the path of the light rays.

An interesting use of the MCMC approach is in determining an estimation of the uncertainty quantification of different models, assimilation schemes, or parameterizations. Van Lier-Walqui et al. [456] applies the MCMC approach to obtain quantification of the uncertainties of the parameterizations of cloud microphysical variables using radar reflectivity. There are some interesting features in the results from [456]; Fig. 22.2 shows a copy of a figure from this paper, where the individual plots are of the posterior joint bivariate marginal distributions. The first feature to notice in Fig. 22.2 is that there are some bivariate distributions that will cause problems for most Gaussian-based data assimilation schemes. Second, if we recall the plots of the bivariate Gaussian, lognormal, and mixed distributions from Chapter 4, then we see that for the unimodal distributions, not many of them appear to be circular or stretched ellipse, which are the signals of an uncorrelated or correlated bivariate Gaussian distribution, respectively. There does appear to be the triangular signal of a uncorrelated bivariate lognormal distribution for some of the parameters, but there also appears to be the signal of the mixed distribution. The information that these plots provide is incredibly important, as it enables us to be aware that developing a Gaussian-based data assimilation, albeit variational, ensemble, or hybrid, will not be optimal for these situations.

In addition to Fig. 22.2, there are several more important figures in [455], which we refer the reader to, so that they may have a fuller understanding of the different situations considered in that paper. However, as noted above, if we were to apply a Gaussian-based data assimilation to this situation, it would be suboptimal and could produce quite bad estimates.

The last application of the MCMC method that we consider before moving onto particle filters is an application of the MCMC approach to quantify the inaccuracies of the analysis error distribution of the ensemble Kalman smoother when applied to obtain the estimates of the cloud microphysical parameters that are close to zero, but cannot become negative, i.e., either positive definite or positive semi-definite. Given the distributions that can be created from the EnKS and the MCMC approach, it

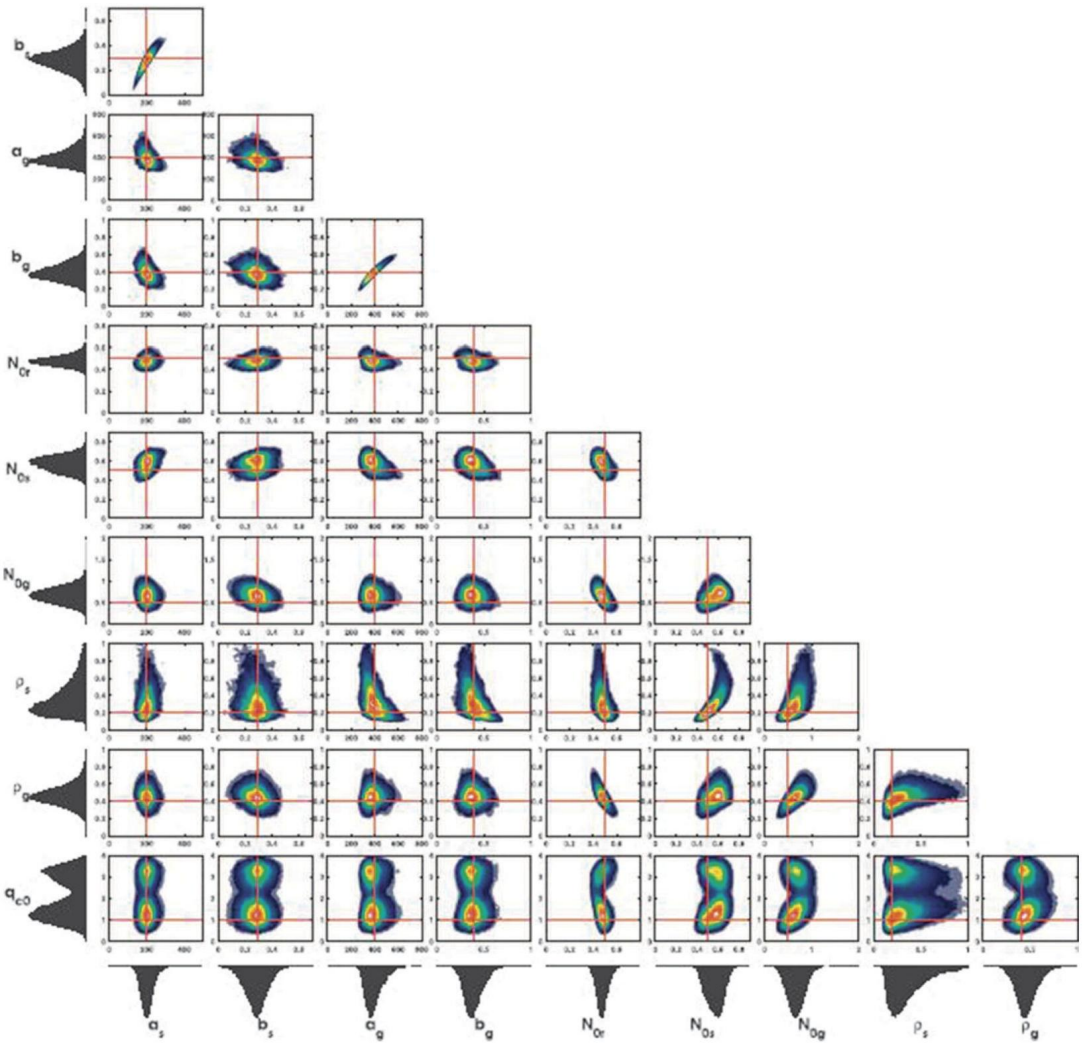


FIGURE 22.2

Copy of a Markov chain Monte Carlo approach to determine relationships between microphysical parameters (Figure 4) from van Lier-Walqui, M., Vukicevic, T., and Posselt, D. J. (2012). Quantification of Cloud Microphysical Parameterization Uncertainty Using Radar Reflectivity, Monthly Weather Review, 140(11), 3442-3466. **© American Meteorological Society. Used with permission.**

is possible to see that the analysis error distribution from the EnKS is "grossly" inaccurate for non-negative microphysical parameters when the parameters are close to zero [343]. Again, this is a way to quantify the sub-optimality of an assumed "Gaussian-fits-all" approach.

22.2 Particle Filters

Particle filters are starting to draw the attention of some operational numerical weather and ocean prediction centers due to the ability of the filters to be applicable for nonlinear and non-Gaussian situations. The drawback of the ensemble Kalman filter (EnKF), which we highlighted before, is that it is based upon the very heavily assumed Gaussian properties of the Kalman filter. If we are applying the extended Kalman filter, we can introduce the nonlinearity of the numerical model and the observation operator, but we are still updating the analysis state and the analysis error covariances as if from a Gaussian distribution.

The MCMC methods do not make the Gaussian assumption for the posterior distribution, and neither do the particle filters. In fact, Mosegaard and Tarantola's 1995 paper refers to particles in the following context:

> Instead of letting p_i represent the probability that a random walker is at point i, we can let p_i represent the number of "particles" at point i.

Particle filters are seen as **sequential MC** techniques [403], which can be shown to produce an ensemble of states drawn from the correct posterior distribution as the ensemble size tends to infinity. The basis of the particle filter is as follows: let $\mathbf{x}$ be the state of a system of dimension N_x, which is a vector containing the discrete approximations of the prognostic variables of the model at a grid point. In [404] the authors state that because it is not possible to determine $\mathbf{x}$ exactly, given imperfect observations, then $\mathbf{x}$ is considered as a random variable, that has a PDF $p(\mathbf{x})$.

We now consider the case that we have a set of observations that enable us to update $p(\mathbf{x})$ as a result of the observations. This, as we have seen, is the posterior distribution. In other words, instead of seeking the mode of $p(\mathbf{x}|\mathbf{y})$, which is what the variational methods do, or the mean, which is what different versions of the ensemble Kalman filters do, we seek the whole PDF at this observation time with the particle filters.

According to van Leeuwen [452], the idea of the particle filter is to try to represent the model PDF by a number of random draws, which are **particles**, or ensemble members. After a previous data assimilation step, a new ensemble of particles is generated. Each particle is propagated forward in time with the **full nonlinear model**. This part of the data assimilation is effectively an approximation to the Fokker-Planck equation introduced in Chapter 20. All of the particle methods have the forward propagation, and they only differ in the way the model and the observations are combined (analysis step).

We now move on to introduce the mathematics of particle filters. The starting point is to introduce the sampling methods. In [452], van Leeuwen introduces the concept of importance sampling, where we have the posterior distribution given by

$$\varrho(\mathbf{x}\,|\,\mathbf{y}) = \frac{\rho(\mathbf{y}\,|\,\mathbf{x})\,\rho(\mathbf{x})}{\int \rho(\mathbf{y}\,|\,\mathbf{x})\,p(\mathbf{x})\,d\mathbf{x}}, \tag{22.14}$$

and the MC approximation to the prior distribution given by

$$p(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x}-\mathbf{x}_i), \tag{22.15}$$

where $\mathbf{x}_i$ is the particle value for $\mathbf{x}$. Now, using the expression in (22.15) in (22.14) results in

$$\varrho\left(\mathbf{x} \mid \mathbf{y}\right) = \sum_{i=1}^{N} w_i \delta\left(\mathbf{x} - \mathbf{x}_i\right), \tag{22.16}$$

where the **weights**, w_i, are given by

$$w_i = \frac{\rho\left(\mathbf{y} \mid \mathbf{x}\right)}{\sum_{j=1}^{N} \rho\left(\mathbf{y} \mid \mathbf{x}_j\right)}. \tag{22.17}$$

The conditional distribution $\rho\left(\mathbf{y} \mid \mathbf{x}\right)$ is often taken to be the Gaussian distribution given by

$$\rho\left(\mathbf{y} \mid \mathbf{x}\right) = A \exp\left\{-\frac{1}{2}\left(\mathbf{y} - \mathbf{h}\left(\mathbf{x}\right)\right)^T \mathbf{R}^{-1}\left(\mathbf{y} - \mathbf{h}\left(\mathbf{x}\right)\right)\right\}. \tag{22.18}$$

The algorithmic description of this particle filter, as given in [452], can be found in Algorithm 22.1.

Algorithm 22.1 Particle Filter

1. Sample N particles $\mathbf{x}_i$ from the initial model probability density $\rho\left(\mathbf{x}^0\right)$.
2. Integrate all particles forward in time up the observation time. This is equivalent to a sample from $\rho\left(\mathbf{x}^1 \mid \mathbf{x}^0\right)$, where this notation can be extended to a general time k.
3. Calculate the weights according to (22.17) and then attach these weights to each corresponding particle.
4. Repeat steps 2 and 3 until all observations up to the present have been *processed*.

Two important features to note about this particle filter are as follows:

1. Balance is preserved in the analysis as the particles are not modified.
2. However, the particles are not modified, only their relative weights; therefore, as the particles move away from the observations, they are not pulled back.

The problem along with a possible remedy here is that after a few analysis steps, one particle receives all the weight, while the other particles have very low weights. This is **filter degeneracy**. In Fig. 22.3, we have a copy of the filter degeneracy plot from [452] to illustrate this problem.

22.2.1 Resampling

As noted, if we apply the sequential importance technique to the particles, we see that the filter collapses to a few particles with large weights, while the remainder tend to a weight of near zero. A class of approaches to try and avoid filter degeneracy are referred to as *resampling*, and we summarize a few techniques here that are presented in [452].

Sequential importance resampling

The idea of resampling is to try to reduce the variance associated with the weights of the particles. There are several techniques for accomplishing this goal. The main motivation for resampling is to abandon

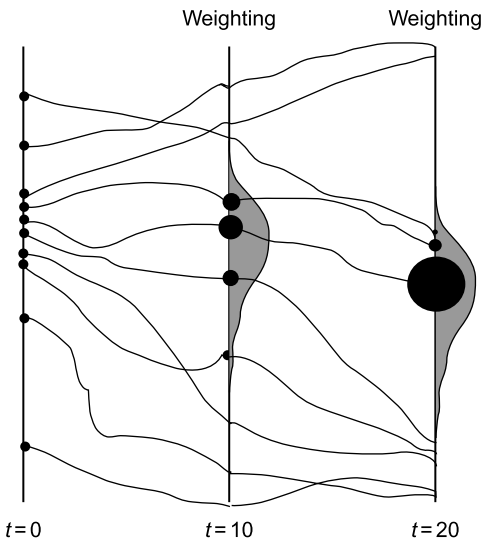


FIGURE 22.3

Copies of the particle filter degeneracy and a possible remedy schematics from (Figure 1) van Leeuwen, P. J. (2009). Particle Filtering in Geophysical Systems, Monthly Weather Review, 137(12), 4089-4114. **© American Meteorological Society. Used with permission.**

the particles with low weights and to replicate the particles with the higher weights; this keeps the number of particles the same. This type of approach is the **sequential importance resampling** or SIR. The idea behind SIR is to replicate particles with the higher weights to go forward to the next analysis time. Then all the particles are given equal weight of $\frac{1}{N}$ to start the next integration. An illustration of the SIR technique from [453] is shown in Fig. 22.4.

Probabilistic resampling

A possible resampling technique is to take random samples directly from the density given by the weights; however, this approach introduces sampling noise.

The basis of the probabilistic resampling appears in Pham [335], where he states that the analysis state of particle filter can be calculated as

$$\mathbf{x}^{a}\left(t_{k}\right)=\sum_{i=1}^{N} w_{i,k}\mathbf{x}_{i}^{f}\left(t_{k}\right), \tag{22.19}$$

and the analysis covariance matrix is given by

$$\mathbf{P^{a}}\left(\mathbf{t_{k}}\right)=\sum_{\mathbf{i=1}}^{\mathbf{N}}\mathbf{w_{j,k}}\left(\mathbf{x_{i}^{f}}\left(\mathbf{t_{k}}\right)-\mathbf{x^{a}}\left(\mathbf{t_{k}}\right)\right)\left(\mathbf{x_{i}^{f}}\left(\mathbf{t_{k}}\right)-\mathbf{x^{a}}\left(\mathbf{t_{k}}\right)\right)^{\mathbf{T}}. \tag{22.20}$$

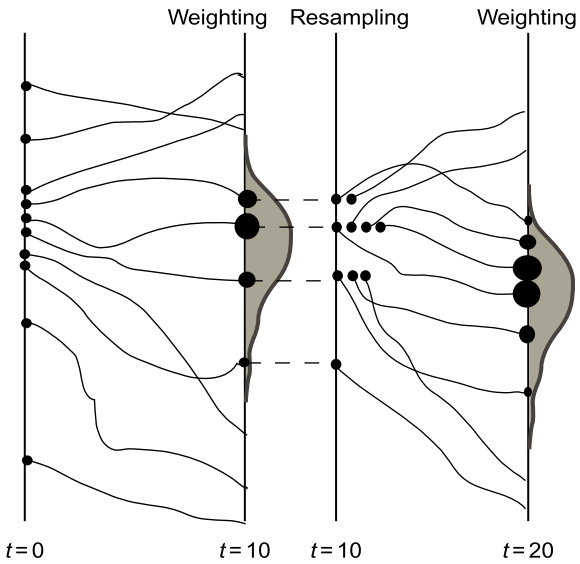


FIGURE 22.4

Copy of the particle filter resampling schematic from (figure 2) van Leeuwen, P. J. (2009). Particle Filtering in Geophysical Systems, Monthly Weather Review, 137(12), 4089-4114. **© American Meteorological Society. Used with permission.**

Therefore, instead of setting $\mathbf{x}_i^a(t_k)$ to $\mathbf{x}_i^f(t_k)$, we draw the particles according to the analysis distribution and reset the weights to $\frac{1}{N}$. However, Pham recommends drawing the particles $\mathbf{x}_1^a(t_k), \ldots, \mathbf{x}_N^a(t_k)$ according to an approximating continuous distribution, where this distribution is suggested to be

$$\varrho\left(\mathbf{x} \mid \mathbf{y}_{0,k}\right) = \sum_{j=1}^{N} w_{i,k}\rho\left(\mathbf{x} - \mathbf{x}_i^f(t_k) \mid h^2\mathbf{P}^{\mathbf{a}}(\mathbf{t_k})\right), \tag{22.21}$$

where $\mathbf{P}^{\mathbf{a}}(\mathbf{t_k})$ is given by (22.20), and h is a small tuning parameter. Therefore, drawing from the density above is the same as drawing for the set $\left\{\mathbf{x}_1^f(t_k), \ldots, \mathbf{x}_N^f(t_k)\right\}$ with weights $w_{1,k}, \ldots, w_{N,k}$, and then adding a Gaussian noise with mean zero and covariance matrix $h^2\mathbf{P}^{\mathbf{a}}(\mathbf{t_k})$.

We should note that as resampling deliberately changes the analysis distribution, from the discrete distribution located at $\mathbf{x}_i^f(t_k)$) with weight $w_{j,k}$ to the one located at $\mathbf{x}_i^a(t_k)$ with weight $\frac{1}{N}$, then this step should only be done if the weights $w_{1,k}, \ldots, w_{N,k}$ differ too much from the uniform distribution.

As a way to measure the discrepancy just mentioned, Pham introduces the entropy difference between the two probability densities as

$$E\left(w_{1,k}, \ldots, w_{N,k}\right) = \log(N) + \sum_{i=1}^{N} w_{i,k}\log w_{i,k}. \tag{22.22}$$

As we can see, E is non-negative and can only be equal to zero when all of the weights are equal to $\frac{1}{N}$. Therefore, resampling should only occur when the entropy is greater than a prescribed threshold. This filter was known as the weighted resampling filter or WRF [212], but that acronym now belongs to the Weather, Research, and Forecasting model.

Residual resampling

Residual resampling is intended to suppress sampling noise. A version of this type of filter is presented in van Leeuwen [451], where it is referred to as the *variance-minimizing filter*. The filter is described as follows. Instead of choosing randomly from the distribution determined by the particle weights, particles with large weights are chosen directly from the distribution in the following manner: first the density is multiplied by the total number of particles. For each of the weights that have been obtained this way, then for those that have a weight larger than one, the integer part of the weight determines the number of identical copies that are made. For example, if a weight was $w_i = 0.105$ and we had 100 particles, then the new weight would be $\hat{w}_i = 10.5$. Thus there would be 10 copies of this particle made, while $\hat{w}_i = 0.5$ remains. Finally, all remaining parts $\hat{w}_i$ form a new density from which the rest of the particles are drawn according to the rule of SIR.

MCMC

As van Leeuwen points out in [451] it is unusual to see the MCMC applied in a particle filter framework; where van Leeuwen is using the Metropolis-Hastings algorithm as a method for determining whether or not to keep a particle, or to replicate the current particle. The algorithm uses the procedures that have already been set out in this chapter to obtain the weights of the particles after they have been integrated in time. The change to the MCMC method is to the acceptance step, which is as follows:

- Choose a particle $\mathbf{x}_i$ as a member of the new ensemble.
- Particle $i + 1$ will be part of the new ensemble if either

$$w_{i+1} > w_i$$

or it is accepted with probability

$$p = \frac{w_{i+1}}{w_i},$$

where this condition is implemented by choosing a random number, ν, from a uniform density on the interval $[0, 1]$, and accepting the particle when $\nu < p$. If $\nu > p$, then the particle is rejected and a replica of $\mathbf{x}_i$ is made.

There are a few more resampling methods presented in [452], and the reader is referred to that paper for more details. However, we have to ask the question: is resampling enough? (The question is also posed in [452].) The reason for asking this is that it is pointed out in [404] that the number of particles required to ensure that filter degeneracy does not occur scales exponentially. This problem is called the **curse of dimensionality** by van Leeuwen [452]. We now consider alternative approach to prevent the degeneracy of the particle filter through changes to the **proposal density**.

22.2.2 Proposal Densities

Since the first edition of this textbook there has been a very good review paper on particle filters in [454], which enables us to improve this section, as well as providing insight into other forms of particle filters.

As stated in [454] the idea case for the particle filter would be to draw independent samples directly from the posterior PDF, as the samples would all have equal weight automatically, where this can only be done when the shape of the posterior PDF is known, and when it is easy to draw from the posterior.

It is stated in [454] that the standard particle filter draws from the prior PDF and that this approach leads to weights that vary too much. However, it is possible to explore the idea of importance sampling of the transition from one time to the next. When the numerical model is not deterministic but stochastic then there is the freedom to change the model equations to move the particles to those parts of the state space to be closer to the observations.

If we assume that we have observations at time n and we have the standard form of Bayes' theorem at time n, then it is possible to write the prior distribution as

$$p\left(\mathbf{x}^n\right) = \int P\left(\mathbf{x} \mid \mathbf{x}^{n-1}\right) p\left(\mathbf{x}^{n-1}\right) d\mathbf{x}^{n-1}, \tag{22.23}$$

where $p\left(\mathbf{x}^n \mid \mathbf{x}^{n-1}\right)$ is referred to as the **transition density**, the PDF of the state at time n, when the state at $n-1$ is known. We know assume that we have a set of weighted particles at time $t = n-1$, with weights w_i^{n-1}. Therefore, we can evaluate (22.23) for the prior as a weighted mixture of transition densities

$$p\left(\mathbf{x}^n\right) \approx \sum_{i=1}^{N} w_i^{n-1} p\left(\mathbf{x}^n \left| \mathbf{x}_i^{n-1} \right.\right). \tag{22.24}$$

Through Bayes' theorem the posterior PDF can be written as

$$p\left(\mathbf{x}^n \left| \mathbf{y}^n \right.\right) \approx \sum_{i=1}^{N} w_i^{n-1} \frac{p\left(\mathbf{y}^n \left| \mathbf{x}^n \right.\right)}{p\left(\mathbf{y}^n\right)} p\left(\mathbf{x}^n \left| \mathbf{x}_i^{n-1} \right.\right). \tag{22.25}$$

Now the prior particles at time n are allowed to arise from following different model equations. Thus we can multiply and divide (22.24) and (22.25) by a **proposal density**, $q\left(\mathbf{x}^n \mid \mathbf{x}^{n-1}, \mathbf{y}^n\right)$, which results in

$$p\left(\mathbf{x}^n\right) \approx \sum_{i=1}^{N} w_i^{n-1} \frac{p\left(\mathbf{x}^n \left| \mathbf{x}_i^{n-1} \right.\right)}{q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)} q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right), \tag{22.26}$$

$$p\left(\mathbf{x}^n \left| \mathbf{y}^n \right.\right) \approx \sum_{i=1}^{N} w_i^{n-1} \frac{p\left(\mathbf{y}^n \left| \mathbf{x}^n \right.\right)}{p\left(\mathbf{y}^n\right)} \frac{p\left(\mathbf{x}^n \left| \mathbf{x}_i^{n-1} \right.\right)}{q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)} q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right). \tag{22.27}$$

It is then stated that drawing from $p\left(\mathbf{x}^n \mid \mathbf{x}^{n-1}\right)$ corresponds to running the original stochastic model. Instead we could draw from $q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)$, that would correspond to a model equation of our choos-

ing. For instance when the original model is given by

$$\mathbf{x}^n = f\left(\mathbf{x}^{n-1}\right) + \boldsymbol{\beta}^n, \tag{22.28}$$

then we can use this as

$$\mathbf{x}^n = g\left(\mathbf{x}^{n-1}, \mathbf{y}^n\right) + \widehat{\boldsymbol{\beta}}^n, \tag{22.29}$$

where $g(\cdot,\cdot)$ is the deterministic part and $\widehat{\boldsymbol{\beta}}^n$ is the stochastic part, that can be freely chosen. In [454] they highlight the fact that $g(\cdot,\cdot)$ is allowed to depend on the observations at the future time, which implies that we generate the prior particles at time n by making one draw from $q\left(\mathbf{x}^n \mid \mathbf{x}^{n-1}, \mathbf{y}^n\right)$ for each i where

$$q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right) = p_{\widehat{\boldsymbol{\beta}}}\left(\mathbf{x}^n - g\left(\mathbf{x}^{n-1}, y^n\right)\right). \tag{22.30}$$

In general, particles at time n are drawn from the alternative model $q\left(\mathbf{x}^n \mid \mathbf{x}^{n-1}, \mathbf{y}^n\right)$ and the weights are changed to account for this. This implies (22.26) and (22.27) become

$$p\left(\mathbf{x}^n\right) = \sum_{i=1}^{N} \widehat{w}_i^{n-1} q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right), \tag{22.31}$$

$$p\left(\mathbf{x}_n \mid \mathbf{y}^n\right) = \sum_{i=1}^{n} \widehat{w}_i^n q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right), \tag{22.32}$$

where the weights here are given by

$$\widehat{w}_i^{n-1} \propto w_i^{n-1} \frac{p\left(x_i n \mid x_i^{n-1}\right)}{q\left(\mathbf{x}_i^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)}, \tag{22.33}$$

and

$$\widehat{w}_i^n \propto \widehat{w}_i^{n-1} \frac{p\left(\mathbf{y}^n \mid \mathbf{x}_i^n\right)}{p\left(\mathbf{y}^n\right)} \propto w_i^{n-1} p\left(\mathbf{y}^n \mid \mathbf{x}_i^n\right) \frac{p\left(\mathbf{x}_i^n \mid \mathbf{x}_{i-1}^n\right)}{q\left(\mathbf{x}_i^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)}. \tag{22.34}$$

As stated in [454], the weights now contain two factors; the likelihood weight, which also appears in the standard particle filter, and a proposal weight. These two weights have opposing effects. If a proposal density strongly pushes the model towards the observations, then the likelihood weight will be large because the difference between observations and model states becomes smaller, but the proposal weight becomes smaller because the model is pushed away from where it wants to go, so $p\left(\mathbf{x}^n \middle| \mathbf{x}_i^{n-1}\right)$ will be small. On the other hand, a weak pushing towards the observations keeps the proposal weight high, but leads to a small likelihood weight. This suggests that there is an **optimum weight** related to an optimal position $\mathbf{x}_i^n$ for each particle as function of its position at time $n-1$.

We now consider some choices for the proposal density.

Relaxation

This form of proposal density is in the form of a relaxation or nudging term to the original equation to steer particles towards the observations and make their weights more similar, [453]. Therefore, we write the model equations now as

$$\mathbf{x}^m = f\left(\mathbf{x}^{m-1}\right) + \mathbf{T}\left(\mathbf{y}^n - \mathbf{h}\left(\mathbf{x}^{m-1}\right)\right) + \widehat{\boldsymbol{\beta}}^m, \tag{22.35}$$

where here the time index m is for the state vector to emphasis that there are several model time steps between observations times, $\mathbf{T}$ is a relaxation matrix to be chosen. In this example the deterministic part consists of the first two terms on the right hand side, while the third term denotes the random part. In [454] they assume that the PDF of the random forcing is Gaussian with mean zero and covariance $\widehat{\mathbf{Q}}$, thus the proposal density is given by

$$q\left(\mathbf{x}^m \mid \mathbf{x}^{m-1}, \mathbf{y}^n\right) = G\left(f\left(\mathbf{x}^{m-1}\right) + \mathbf{T}\left(\mathbf{y}^n - \mathbf{h}\left(\mathbf{x}^{m-1}\right)\right), \widehat{\mathbf{Q}}\right), \tag{22.36}$$

since the PDF of $\mathbf{x}^m$ is a shift in the mean of the PDF of $\widehat{\boldsymbol{\beta}}^m$. It is assumed that for the original model the random part is Gaussian with zero mean and covariance $\mathbf{Q}$, so that

$$p\left(\mathbf{x}^m \mid \mathbf{x}^{m-1}\right) = G\left(f\left(\mathbf{x}^{m-1}\right), \mathbf{Q}\right). \tag{22.37}$$

The change in the model equations is compensated for in the particle filter buy a change in the relative weights of each particle by

$$w_i^m = w_i^{m-1} \frac{p\left(\mathbf{x}_i^m \mid \mathbf{x}_i^{m-1}\right)}{q\left(\mathbf{x}_i^m \mid \mathbf{x}_i^{m-1}, \mathbf{y}^n\right)} \propto w_i^{m-1} \frac{\exp\left\{-J_p\right\}}{\exp\left\{-J_q\right\}}, \tag{22.38}$$

where for Gaussian errors

$$J_p = \frac{1}{2}\left(\mathbf{x}_i^m - f\left(\mathbf{x}_i^{m-1}\right)\right)^T \mathbf{Q}^{-1}\left(\mathbf{x}_i^m - f\left(\mathbf{x}_i^{m-1}\right)\right), \tag{22.39}$$

$$\begin{aligned} J_q &= \frac{1}{2}\left(\mathbf{x}_i^m - f\left(\mathbf{x}_i^{m-1}\right) - \mathbf{T}\left(\mathbf{y}^n - \mathbf{h}\left(\mathbf{x}^{m-1}\right)\right)\right)^T \widehat{\mathbf{Q}}^{-1}\left(\mathbf{x}_i^m - f\left(\mathbf{x}_i^{m-1}\right) - \mathbf{T}\left(\mathbf{y}^n - \mathbf{h}\left(\mathbf{x}^{m-1}\right)\right)\right), \\ &= \frac{1}{2}\left(\widehat{\boldsymbol{\beta}}_i^m\right)^T \widehat{\mathbf{Q}}^{-1} \widehat{\boldsymbol{\beta}}_i^m. \end{aligned} \tag{22.40}$$

Weighted Ensemble Kalman Filter (WEKF)

In [454] they state that it is possible to other existing data assimilation methods in proposal densities, like EnKFs or variational methods. In the Weighted Ensemble Kalman Filter (WEKF), [327], the stochastic EnKF from [53] is used as follows. The EnKF update can be written as:

$$\mathbf{x}_i^n = \mathbf{x}_i^f + \mathbf{K}\left(\mathbf{y} - \mathbf{H}\mathbf{x}_i^f - \boldsymbol{\epsilon}_i\right), \tag{22.41}$$

where $\mathbf{x}_i^f = f\left(\mathbf{x}_i^{n-1}\right) + \widehat{\boldsymbol{\beta}}_i^n$, and substituting this expression in the equation above yields

$$\mathbf{x}_i^n = \mathbf{x}_i^f + \mathbf{K}\left(\mathbf{y} - \mathbf{H} f\left(\mathbf{x}_i^{n-1}\right)\right) + (\mathbf{I} - \mathbf{K}\mathbf{H})\,\widehat{\boldsymbol{\beta}}_i^n - \mathbf{K}\boldsymbol{\epsilon}_i, \tag{22.42}$$

where in terms of the sun of the deterministic and stochastic parts can be written as

$$\mathbf{e}^n = g\left(\mathbf{x}^{m-1}, \mathbf{y}^n\right) + \widehat{\boldsymbol{\beta}}_i^n. \tag{22.43}$$

Thus the proposal density here is

$$q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right) = G\left(f\left(\mathbf{x}^{n-1}\right) + \mathbf{K}\left(\mathbf{y}^n - \mathbf{H}f\left(\mathbf{x}^{n-1}\right)\right), \widehat{\mathbf{Q}}\right), \tag{22.44}$$

with

$$\widehat{\mathbf{Q}} \equiv (\mathbf{I} - \mathbf{KH})\,\mathbf{Q}\,(\mathbf{I} - \mathbf{KH})^2 + \mathbf{KRK}^T. \tag{22.45}$$

22.2.3 Optimal Proposal Density

In [403], Snyder introduces the terms "standard" and "optimal" proposals to represent the choice for the proposal distribution, where a cautionary note made in [403], is that the term "optimal" does not refer to the performance of the resulting particle filter; instead it refers to the fact that this proposal distribution minimizes the variance of the weights over different random draws of $\mathbf{x}_i^k$.

However, in [405] Synder et al. provide a proof of this optimality, and in [454] they generalize this result and show that the optimal proposal density is optimal even when each particle has its own proposal density that is allowed to depend on all previous particles i.e. $q\left(\mathbf{x}^n \mid i, \mathbf{x}_{i:N}^{n-1}, \mathbf{y}^n\right)$.

[405] concentrate on the optimal representation of $p\left(x_n, x_{n-1} \mid y^n\right)$ in a sequential algorithm and introduce the random variable

$$w^*\left(\mathbf{x}^n, \mathbf{x}^{n-1}\right) = \frac{p\left(\mathbf{x}^n, \mathbf{x}^{n-1} \mid \mathbf{y}^n\right)}{q\left(\mathbf{x}^n, \mathbf{x}^{n-1} \mid \mathbf{y}^n\right)}, \tag{22.46}$$

and determine that proposal density q that minimizes the variance in the weights w^*, with the expectation taken over the density that the particles are drawn from; proposal q.

In [454] they provide a proof that the optimal proposal density is also optimal for the strict filtering case, so when we are interested in minimal variance of the weights at time n only. Specifically, the question they raise is: given the set of particles at $t = n-1$ drawn from $p\left(\mathbf{x}_{n-1} \mid \mathbf{y}_{1:n-1}\right)$, which proposal density of the form $q\left(\mathbf{x}^n \middle| i, \mathbf{x}_{1:N}^{n-1}, \mathbf{y}^n\right)$ gives minimal variance of the weights at time n? We refer the reader to [454] for the full proof and discussions for Gaussian distributions.

22.2.4 Implicit Particle Filter

In [454] they introduce the implicit particle filter as an indirect way to draw from the optimal proposal, even over several time steps. The assumption that the model errors of both the model and proposal density are Gaussian and the observation operator $\mathbf{H}$ is linear. However, when we have the nonlinear observation operator $\mathbf{h}$, or when the proposal density is used over several model time steps, the density to draw from is not now Gaussian. In [61] they realized that it is possible to draw from a Gaussian and then apply a transformation to that draw to find samples from the optimal proposal density. We have a copy of the schematic of the implicit particle filter from [454] in Fig. 22.5

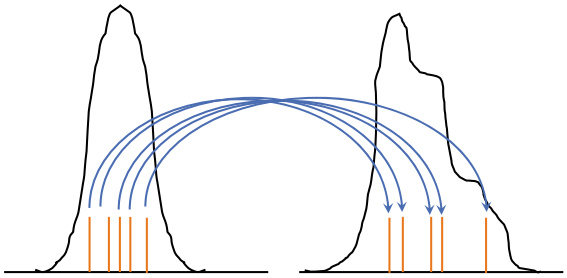


FIGURE 22.5

Copy of figure 4 from Van Leeuwen, PJ, Künsch, HR, Nerger, L, Potthast, R, Reich, S. Particle filters for high-dimensional geoscience applications: A review. Q J R Meteorol Soc. 2019; 145: 2335–2365. https://creativecommons.org/licenses/by/4.0/.

We start by recalling that the posterior PDF can be written as

$$p\left(\mathbf{x}^n \mid \mathbf{y}^n\right) = \sum_{i=1}^{N} w_i \frac{p\left(\mathbf{y}^n \mid \mathbf{x}^n\right)}{p\left(\mathbf{y}^n\right)} \frac{p\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}\right)}{q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right)} q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right). \tag{22.47}$$

This scheme draws from a Gaussian proposal $q\left(\boldsymbol{\xi}\right) = G\left(\mathbf{0}, \mathbf{I}\right)$, and the transformation can be written as $q\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}, \mathbf{y}^n\right) = q\left(\boldsymbol{\xi}\right) \mathbf{J}_i^{-1}$, where $\mathbf{J}$ is the Jacobian of the transformation from $\mathbf{x}^n$ to $\boldsymbol{\xi}$, where this transformation is found implicitly, hence the name, by defining

$$F_i\left(\mathbf{x}^n\right) = -\log\left(p\left(\mathbf{y}^n \mid \mathbf{x}^n\right) p\left(\mathbf{x}^n \mid \mathbf{x}_i^{n-1}\right)\right), \tag{22.48}$$

and, after drawing $\boldsymbol{\xi}_i$ for each particle, solving for $\mathbf{x}^n$ in

$$F_i\left(\mathbf{x}^n\right) = \frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi} + \boldsymbol{\phi}_i, \tag{22.49}$$

for each particle, where $\boldsymbol{\phi}_i = \min_{x^n} F_i\left(\mathbf{x}^n\right) \propto p\left(\mathbf{y}^n \mid \mathbf{x}_i^{n-1}\right)$. Thus the weights of the particles can be shown to be

$$w_i^n = w_i^{n-1} \exp\left\{-\boldsymbol{\phi}_i\right\} \mathbf{J}_i. \tag{22.50}$$

The ability to solve [454] is not straightforward, and the reader is referred to [308] for a suggestion to do so.

There are a couple of approaches for the optimal density that are presented in [454]; equal weights bu resampling at time $n-1$, equivalent weights along with implicit equal weights, and the reader is referred here to see their details.

22.2.5 Transportation Particle Filters

In resampling particle filters, the prior particles are first weighted to represent the posterior and then transformed to unweighted particles through duplicating high-weight particles and abandoning low-weight particles. In transformation particle filters, a transformation is sought that moves particles from the prior to particles of the posterior in a deterministic manner. A related approach, that uses random transformation steps, is based on tempering the likelihood, [454].

The first thing we consider here is an one-step transportation where we attempt to transform samples from the prior into samples from the posterior in one transformation. An example of this approach is the Ensemble Transform Particle Filter, [363], in which unweighted particles are linear combinations of the weighted particles, such that

$$\mathbf{X}^a = \mathbf{X}^f \mathbf{D}, \tag{22.51}$$

in which the matrices $\mathbf{X}^{f,a} \equiv \left(\mathbf{x}_1^{f,a}, \mathbf{x}_1^{f,a}, \ldots, \mathbf{x}_N^{f,a}\right)$ and $\mathbf{D}$ is a transformation matrix. The only conditions on $\mathbf{D}$ are that $d_{ij} \geq 0$, $\sum_i d_{ij} = 1$, and $\sum_j d_{ij} = w_i N$. These three conditions leave a lot of freedom for all N^2 elements of $\mathbf{D}$, and a useful way to determine them is to ensure minimal overall movement in state space of the particles from prior to posterior. This leads to an optimal transportation problem and is typically solved by minimizing a cost function that penalizes movement of particles. See references in [454] for more details.

22.2.6 Tempering of the Likelihood

Instead of trying to transform the particles from the prior to particles from the posterior in one step, one can also make this a smoother transition. In [454] they refer to tempering to achieve this where one factorizes the likelihood as follows:

$$p\left(\mathbf{y}|\mathbf{x}\right) = p\left(\mathbf{y}|\mathbf{x}\right)^{\gamma_1} p\left(\mathbf{y}|\mathbf{x}\right)^{\gamma_2} \ldots p\left(\mathbf{y}|\mathbf{x}\right)^{\gamma_m}, \tag{22.52}$$

with $0 < \gamma_i < 1$ and ensuring that the sum of the γs is equal to 1. Then the weighting of the particle filter is first done with the first factor

$$p_1\left(\mathbf{x} \mid \mathbf{y}\right) = \frac{p\left(\mathbf{y}|\mathbf{x}\right)^{\gamma_1}}{p\left(\mathbf{y}\right)^{\gamma_1}} p\left(\mathbf{x}\right). \tag{22.53}$$

It is stated in [454] that the reason for this is that the likelihood is much less peaked, and hence the degeneracy can be avoided when γ_1 is small enough. To illustrate this we have a copy of Figure 6 from [454] in Fig. 22.6.

The particles are resampled, and now the weighting is performed using the second factor, followed by resampling, etc. In this way the scheme slowly moves all particles towards the high-probability regions of the posterior. Of course, after resampling several particles will be identical, so one needs to jitter the particles, so perturbing them slightly, to regain diversity. See [454] for a full explanation about this procedure.

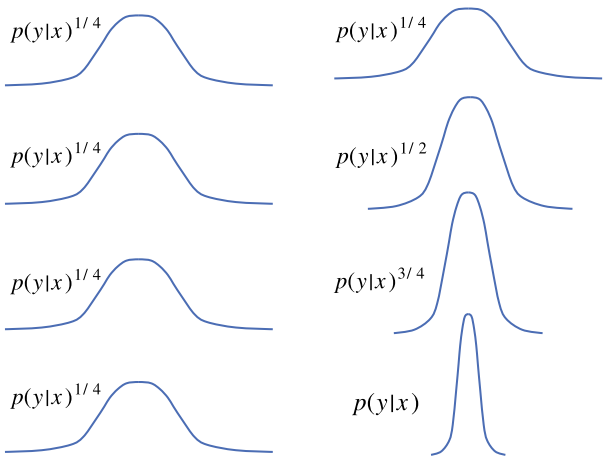


FIGURE 22.6

Copy of figure 6 from Van Leeuwen, PJ, Künsch, HR, Nerger, L, Potthast, R, Reich, S. Particle filters for high-dimensional geoscience applications: A review. Q J R Meteorol Soc. 2019; 145: 2335–2365. https://creativecommons.org/licenses/by/4.0/.

22.2.7 Particle Flow Filters

A new approach that is gaining steam are the schemes that dynamically move the particles in state space from equal-wright particles represent the prior $p(\mathbf{x})$ to equal-weight particles representing the posterior $p(\mathbf{x}|\mathbf{y})$, and so we seek a differential equation

$$\frac{d}{ds}\mathbf{x} = \mathbf{f_s}(\mathbf{x}), \tag{22.54}$$

in artificial time $s \geq 0$ with the flow map defining the desired transformation. If the initial conditions of (22.54) are chosen from a PDF $p_0(\mathbf{x})$, then the solutions follow a distribution characterized by the Liouville equation

$$\frac{\partial p_s}{\partial s} = -\nabla_{\mathbf{x}} \cdot (p_s \mathbf{f_s}), \tag{22.55}$$

with initial condition $p_0(\mathbf{x}) = p(\mathbf{x})$ and final condition $p_{s_{final}}(\mathbf{x}) = p(\mathbf{x}|\mathbf{y}^n)$.

In [454] they indicate that two classes of particle flow filters arise. With the first class we start from the tempering approach, such that $s_{final} = 1$. We now take the limit of more and more tempering steps by choosing $\gamma_i = \frac{1}{n} = \Delta s$ with $\lim_n \to \infty$, so $\lim_{\gamma_i \to 0}$ or $\lim_{\Delta s \to 0}$. This leads to

$$\begin{aligned}
\lim_{\Delta s \to 0} p_{s+\Delta s}(\mathbf{x}) &= p_s(\mathbf{s}) \left(\frac{p(\mathbf{y}|\mathbf{y})}{p(\mathbf{y})} \right)^{\Delta s}, \\
&= p_s(\mathbf{x}) \exp\{\Delta s (\log p(\mathbf{y}|\mathbf{x}) - \log p(\mathbf{y}))\}, \\
&\approx p_s(\mathbf{x}) (1 - \Delta s \log p(\mathbf{y}|\mathbf{x}) - \Delta s \log p(\mathbf{y})).
\end{aligned} \tag{22.56}$$

Hence we find

$$\frac{\partial p_s(\mathbf{x})}{\partial x} = -\nabla_{\mathbf{x}} \cdot (p_s \mathbf{f_s}) = p_s(x)(\log p(\mathbf{y} \mid \mathbf{x}) - c_s), \tag{22.57}$$

with $c_s \equiv \int p_s(\mathbf{x}) \log p(\mathbf{y} \mid \mathbf{x}) d\mathbf{x}$. In [454] that state that explicit expressions for fs are available for certain PDFs such as Gaussians and Gaussian mixtures. These particle flow filters can be viewed as a continuous limit of the tempering methods described in the previous subsection, avoiding the need for resampling and jittering. We have a copy of figure 7 from [454] that is a schematic of the steps in the particle flow filter in Fig. 22.7. Alternatively, we could explore ideas from MCMC. One MCMC method that is mentioned in [454] that generates samples from the posterior is the Langevin Monte-Carlo sampling, in which a sequence of samples is generated by

$$x^{j+1} = x^j - \Delta s \nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) + \sqrt{2\Delta s}\beta^j, \tag{22.58}$$

in which β^j is a random forcing term drawn from $G(0, I)$ The corresponding Fokker–Planck equation for this stochastic PDE is given by

$$\begin{aligned} \frac{\partial p_s}{\partial s} &= \nabla \cdot (p_s \nabla(-\log p(\mathbf{x} \mid \mathbf{y}))) + \nabla_{\mathbf{x}} \cdot \nabla_{\mathbf{x}} \log p_s, \\ &= -\nabla_{\mathbf{x}} \cdot (p_s(\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) - \nabla_{\mathbf{x}} \log p_s)). \end{aligned} \tag{22.59}$$

This equation corresponds to the deterministic partial differential equation in (22.54), where $\mathbf{f_s}(\mathbf{x})$ is given by:

$$\mathbf{f}_s(\mathbf{x}) = p_s \{\nabla_{\mathbf{x}}(p_s(\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) - \nabla_{\mathbf{x}} \log p_s))\}.$$

[454] goes into a lot more details about the MCMC approach and the reader is referred there for more details. The paper also goes on to review the localized PF, the localized adaptive PF, Local Ensemble transform PF, the different space-time filters, as well as hybrids between particle filters and ensemble Kalman filters, as well as the merging PF, along with nonlinear ensemble transform filter and a hybrid LETPF-LETKF and the nonlinear ensemble adjustment filter.

However, in the next section we review one of the localized particle filters, specifically the local particle filter from [345], that is gaining a lot of attention as it tests the ability to be operationally viable.

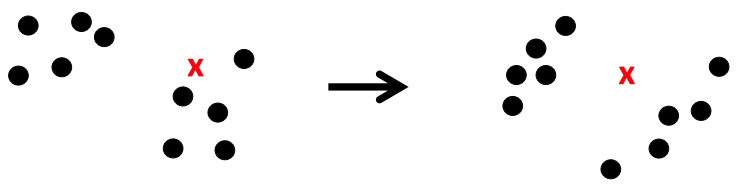


FIGURE 22.7

Copy of figure 7 from Van Leeuwen, PJ, Künsch, HR, Nerger, L, Potthast, R, Reich, S. Particle filters for high-dimensional geoscience applications: A review. Q J R Meteorol Soc. 2019; 145: 2335–2365, https://creativecommons.org/licenses/by/4.0/.

22.3 Local Particle Filter

As we have seen with the ensemble based data assimilation methods, we need a form of localization first to stop observational influence spreading too far, but also to reduce the impact of sampling error from the size of the ensemble. Here localization is proposed in [345] as a method to try and prevent filter degeneracy. To describe how localization may be achieved in a PF framework, [345] starts by considering the case where a single observation, y, is available and posterior weights are calculated using the likelihood of the observation given each particle.

One means of achieving localization is to extend the original weights from scalars to vectors of length N_x, that are denoted $\boldsymbol{\omega}_n$. The resulting vectors form the columns of a $N_x \times N_e$ weighting matrix, and are constructed to reflect the local influence of observations on the posterior estimate. As in EnKFs, the influence of observations on neighboring state-space updates is specified using prior knowledge of the *physics* of the system (e.g., physical length scales contributing to spatial correlations in the prior). This form of localization is achieved by including the function, $l\left[y, x_j, r\right]$, in the calculation of the j-th elements of each $\boldsymbol{\omega}_n$ and their normalization vector $\boldsymbol{\Omega}$:

$$\omega_{n,j} = \left[p\left(y \mid x_{n,j}\right) - 1\right] l\left[y, x_j, y\right] + 1, \tag{22.60}$$

$$\Omega_j = \sum_{m=1}^{N_e} \omega_{m,j}. \tag{22.61}$$

In [345] the localization function has a maximum value of 1 when the Euclidean distance between y and x_j is 0, and decays to 0 when y and x_j are far apart; the rate of this decay is controlled by the parameter r. In practice, $l\left[y, x_j, r\right]$ should be a smooth function with compact support. In the work that is presented in [345] they use the function from (4.10) in [152] for $l\left[y, x_j, r\right]$, that has a Gaussian-type structure with a width specified by r.

In [345] it is stated that the equation chosen for forming the vector weights is motivated by two factors. The first advantage of (22.60) is that it localizes information spatially. The normalized weights $\boldsymbol{\omega} \oslash \boldsymbol{\Omega}$ reflect the observation likelihood near y and the prior weights $\frac{1}{N_e}$ away from y. The second motivating factor mentioned involves the computation of weights when given multiple observations. Assuming observation errors are independent, $p\left(\mathbf{y} \mid \mathbf{x}_b\right)$ can be written as $\prod_{i=1}^{N_y} p\left(y_i \mid \mathbf{x}_n\right)$, where y_i is the i-th observation in $\mathbf{y}$. The values for the j-th elements of the weights, given the i-th observation are then found sequentially by

$$\begin{aligned}\omega_{n,j}^{(y_i)} &= \prod_{q=1}^{i} \left\{\left[p\left(y_q \middle| x_{n,j}^{(y_0)}\right) - 1\right] l\left[y_q, x_j, r\right] + 1\right\}, \\ &= \omega_{n,j}^{(y_{i=1})} \left\{\left[p\left(y_q \middle| x_{n,j}^{(y_0)}\right) - 1\right] l\left[y_q, x_j, r\right] + 1\right\},\end{aligned} \tag{22.62}$$

$$\Omega_j^{(y_i)} = \sum_{n=1}^{N_e} \omega_{n,j}^{(y_i)}, \tag{22.63}$$

where superscript (y_i) refers to quantities that reflect all observations up to y_i and $\mathbf{x}^{(y_0)}$ is the prior ensemble before assimilating any observations in $\mathbf{y}$. It is then mentioned in [345] that for applications where many observations are assimilated over a large spatial domain, most values in the product of

(22.62) will be equal to 1. The resulting weight equation is numerically stable for large N_y, because the rate at which this product approaches zero depends only on the number of observations within the localization region defined by $l\left[y_q, x_j, r\right]$. After applying (22.62) and (22.63) to calculate the weights, posterior quantities are approximated using

$$\overline{f(\mathbf{x})} \approx \sum_{n=1}^{N_e} (\boldsymbol{\omega}_n \oslash \boldsymbol{\Omega}) \circ f(\mathbf{x}), \tag{22.64}$$

where f represents the posterior PDF here.

Given this localization method just presented the local particle filter, or LPF, as it is often referred to, is based upon two stages: sampling and merging, and a probability mapping stage. The problem that arises is the ability to generate equally likely samples from the posterior density, where a typical sampling strategy for low-dimensional stochastic systems is to remove particle with small weights and duplicate particles with large wrights (i.e. bootstrap filter), and a similar approach is applied in the LPF, but we should be aware that localization adds complexity to the process because a unique weight exists for each element of the state vector. In [345] the approach that is taken is to process observations at each filter time serially, while recursively updating particles. Thus the two steps are: 1) apply bootstrap resampling for each observation and merge prior particles with resampled particles to generate samples from a distribution with the approximate first- and second-order moments; and 2) use probability mapping to adjust the new particles so that they are consistent with the marginal probabilities given by the set of posterior weights for each variable. An additional objective of the first step is to preserve the sampled particles near each observation, so that the updated particles approach the bootstrap filter solution near observations. The second step provides higher order corrections to the particles not considered during the first step.

We now summarize the two steps of the LPF; **sampling and merging step**: We start be considering the adjustment of particles associated with the i-th observation. The prior error distribution before assimilating y_i is approximated with N_e equally likely particles that represent samples from the probability density, given all observations up to y_{i-1}; denoted by $\mathbf{x}_n^{(y_{i-1})}$ for $n = 1, 2, \ldots, N_e$. To maintain consistency with the localized weighting vectors, the LPF has to create posterior particles that satisfy the Bayesian solution in regions of the state space assumed to be influenced by y_i. Whereas in regions of the state space that are assumed to be independent of y_i must maintain characteristics of the prior. To achieve this result, a scalar weight $\widetilde{w}_n = p\left(y_i \,|\, \mathbf{x}_n^{(y_{i-1})}\right)$ is first calculated for each particle, and then normalized by $\widetilde{W} = \sum_{n=1}^{N_e} \widetilde{w}_n$. These weights are then used to sample N_e particles with replacement to provide posterior particles that would result from applying the bootstrap filter. Updates are then made to the prior particles in a manner that is consistent with the bootstrap filter solution near observations, and the first two moments of the localized posterior solution in the neighborhood of the observations:

$$\mathbf{x}_n^{(y_i)} = \overline{\mathbf{x}}^{(y_i)} + \mathbf{r}_1 \circ \left(\mathbf{x}_{k_n}^{(y_{i-1})} - \overline{\mathbf{x}}^{(y_i)}\right) + \mathbf{r}_2 \circ \left(\mathbf{x}_n^{(y_{i-1})} - \overline{\mathbf{x}}^{(y_i)}\right), \tag{22.65}$$

where $\overline{\mathbf{x}}^{(y_i)}$ is the posterior mean calculated by (22.64) and k_n is the index of the n-th sampled particle.

According to [345] the new particles are formed as linear combinations of the sampled particles and prior particles using the coefficient vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ of length N_x to specify the influence of localization

on the updates. The form chosen for (22.65) provides a means of deriving an update equation that satisfies the bootstrap filter solution at the location of observations, and the posterior mean and variance calculated from (22.64) within the localization region. Therefore the j-th elements of $\mathbf{r}_1$ and $\mathbf{r}_2$ are show in [345] to be

$$r_{1,j} = \sqrt{\frac{\sigma_j^{(y_i)^2}}{\frac{1}{N_e - 1}\sum_{n=1} N_e \left(x_{k_n}^{(y_{i-1})} - \overline{x}_j^{(y_i)} + c_j \left(x_{k_n}^{(y_{i-1})} - \overline{x}_j^{(y_i)} \right) \right)^2}}, \tag{22.66}$$

$$r_{2,j} = c_j r_{1,j}, \tag{22.67}$$

$$c_j = \frac{N_e \left(1 - l\left[x_j, y_i, r\right]\right)}{l\left[x_j, y_i, r\right] \widetilde{W}}, \tag{22.68}$$

where $\sigma_j^{(y_i)^2}$ is the error variance conditioned on all observations up to y_i. A comment is made at this point in [345] that posterior correlations between state variables are not considered during this formulation, but are provided implicitly through the sampling step of the algorithm.

We now consider the **probability mapping step**: From [345] after updating the particles within the localization region using (22.65), higher-order corrections are then made using the probability mapping methods kernel density distribution mapping (KDDM) approach. KDDM operates by mapping a prior sample into a posterior sample that matches the quantiles of a specified posterior distribution, for the LPF [345] states that the desired posterior distribution is defined by the prior particles and their posterior weights. One advantage of KDDM is that when applied separately for each state variable in $\mathbf{x}$, the resulting posterior ensemble contains approximately the same correlations as the prior ensemble, [345]. Therefore, univariate KDDM steps can be applied to the particles while maintaining the cross-variable correlations that resulted from the sampling part of the update algorithm. Below is a copy of the description of the KDDM where Poterjoy has denotes the j-th values of input (prior) and output (posterior) particles as $x_{n,j}^f$ and $x_{n,j}^a$, respectively.

Starting from the recently updated particles and weights, KDDM uses the following steps to perform the mapping [345]:

1. Approximate the prior and posterior densities using linear combinations of Gaussian kernels. This step uses a sum of kernels centered on each $x_{n,j}^a$, that are weighted by $\frac{1}{N_e}$ to form a prior PDF (PDF^f) and $\frac{\omega^{(y_i)}}{\Omega_j^{(y_i)}}$ to form a posterior PDF (PDF^a).
2. Integrate the two PDFs numerically via the trapezoid rule to form the prior CDF (CDF^f) and posterior CDF (CDF^a).
3. Apply cubic spline interpolation of each point member: $c_{n,j}^f = \text{CDF}^f\left(x_{n,j}^f\right)$.
4. Estimate posterior particles by applying cubic spline interpolation to the find the inverse of the posterior CDF at $c_{n,j}^f$: $x_{n,j}^a = (\text{CDF}^a)^{-1}\left(c_{n,j}^f\right)$.

There has been much traction with the LPF since [345], and it had a major update in [346], and we refer the reader there for these updates. It is also showing much promise towards operations as well.

22.4 Particle Smoother

In this section we shall present the work from [307] where they present the variational particle smoother, motivated by the success of 4D VAR and the use of particle filters inside of 4D VAR. In [307] they sate that 4D VAR methods work with the posterior distribution $p\left(\mathbf{x}_{k-1} | \mathbf{y}_{1:k}\right)$, where particle filters usually work with $p\left(\mathbf{x}_k \mid \mathbf{y}_{1:k}\right)$, and $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k}\right)$ so there appears to be a connection between smoothers and 4D VAR. They define a particle smoother as a sampling method for $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k}\right)$. It is possible to use results from $k-1$ by using the numerical model to evolve the smoother-ensemble to observation time, as is the case in 4D VAR.

Therefore, particle smoothers work as follows: We select a proposal distribution $q\left(\mathbf{x}_{k-1}; \mathbf{y}_{1:k}\right)$ draw samples from it, and then attach to each sample a weight and then attach to each sample a weight

$$w \propto \frac{p\left(\mathbf{y} \mid \mathbf{x}_{k-1}\right) p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right)}{q\left(\mathbf{x}_{k-1}; \mathbf{y}_{1:k}\right)}. \tag{22.69}$$

The weighted ensemble $\left\{\mathbf{x}_{k-1}^j, w^j\right\}, \quad j = 1, 2, \ldots, N$ approximates the posterior distribution $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k}\right)$ in the sense that weighted averages over the ensemble coverage to expected values with respect to the posterior distribution $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k}\right)$ as $N_e \rightarrow \infty$. However, in practice we run into the problem that these weights cannot be evaluated, because $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right)$ in the numerator of (22.69) is generally not known

The above deficiency can be overcome by using a Gaussian approximation for $p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right)$, in the definition from [307] for the posterior distribution

$$p\left(\mathbf{x}_{k-1} | \mathbf{y}_{1:k}\right) \propto p\left(\mathbf{y}_k \mid \mathbf{x}_{k-1}\right) p\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right) \tag{22.70}$$

and replace it in the expression above with $\widetilde{p}\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right) = G\left(\boldsymbol{\mu}, \mathbf{B}\right)$ to then be able to construct the approximate posterior distribution $\widehat{p}$ in

$$\widehat{p}\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{i:k-1}\right) \propto \widetilde{p}\left(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}\right) p\left(\mathbf{y}_k \mid \mathbf{x}_{k-1}\right). \tag{22.71}$$

In [307] they define the 4D VAR cost function as

$$\begin{aligned} J\left(\mathbf{x}_{k-1}\right) = & \frac{1}{2}\left(\mathbf{x}_{k-1} - \boldsymbol{\mu}\right)^T \mathbf{B}^{-1}\left(\mathbf{x}_{k-1} - \boldsymbol{\mu}\right) \\ & + \frac{1}{2}\left(\mathbf{h}\left(f\left(\mathbf{x}_{k-1}\right)\right) - \mathbf{y}_k\right)^T \mathbf{R}^{-1}\left(\mathbf{h}\left(f\left(\mathbf{x}_{k-1}\right)\right) - \mathbf{y}_k\right), \end{aligned} \tag{22.72}$$

where $f\left(\mathbf{x}\right)$ is the deterministic model. Thus Morzfeld et al. suggests that a natural choice for a proposal distribution is the Gaussian

$$\begin{aligned} q\left(\mathbf{x}_{k-1}; \mathbf{y}_{1:k}\right) &= G\left(\mathbf{x}^*, \mathbf{G}^{-1}\right), \\ &\propto \exp\left\{-\frac{1}{2}\left(\mathbf{x}_{k-1} - \mathbf{x}^*\right)^T \mathbf{G}\left(\mathbf{x}_{k-1} - \mathbf{x}^*\right)\right\}, \end{aligned} \tag{22.73}$$

where $\mathbf{x}^*$ is the minimizer for the cost function above, and $\mathbf{G}$ is the approximate Hessian of the cost function, evaluated at the minimizer. With this proposed distribution, the weights become

$$w \propto \frac{p\left(\mathbf{y}_k \mid \mathbf{x}_{k-1}\right) \widetilde{p}\left(\mathbf{x}_{k-1} \mid \mathbf{y}1 : k-1\right)}{q\left(\mathbf{x}_{k-1}; \mathbf{y}_{1:k}\right)},$$

$$\propto \frac{\exp\left\{-J\left(\mathbf{x}_{k-1}\right)\right\}}{\exp\left\{-\frac{1}{2}\left(\mathbf{x}_{k-1}-\mathbf{x}^*\right)^T \mathbf{G}\left(\mathbf{x}_{k-1}-\mathbf{x}^*\right)\right\}}. \tag{22.74}$$

To avoid under- or overflow, it is suggested to consider computing the negative logarithm of the weights, $\widehat{w} = -\log(w)$. Once all N_e negative-log weights $\widehat{w}$ are computed, it is possible to subtract their minimum value from all of them, then take the exponential, then normalize so that the weights sum to one.

The weighted ensemble $\left\{\mathbf{x}_{k-1}^j, w^j\right)\ j = 1, 2, \ldots, N$ approximates the posterior distribution $\widehat{p}$ in (22.71). Generating samples in this way is an implementation of the implicit sampling from the implicit particle filter. We should note that although the proposal density here is Gaussian, the associated posterior is not necessarily a Gaussian distribution.

We have a recreation of the algorithmic description from [307] in Algorithm 22.2, where the localization that is mentioned here, is explained in detail in this reference and the reader is referred to here for them.

Algorithm 22.2 Copy of the Variational particle smoother (varPS)

Solve the variational problem: minimize $j\left(\mathbf{x}^{n-1}\right)$,
Result: minimizer $\mathbf{x}^*$ and Hessian $\mathbf{G}$
Localize/inflate proposal covariance $\mathbf{G}^{-1}$
Sampling: draw an ensemble of N_e particles from the proposal: $\mathbf{x}_{k-1}^j\ G\left(\mathbf{x}^*, \mathbf{G}^{-1}\right)$
Compute and store the corresponding states at time k (running the model N_e times)
for $j = 1, 2, \ldots, N_e$ **do**

$$\text{Compute weights: } w_j \propto \frac{\exp\left\{-J\left(\mathbf{x}_{k-1}\right)\right\}}{\exp\left\{-\frac{1}{2}\left(\mathbf{x}_{k-1}-\mathbf{x}^*\right)^T \mathbf{G}\left(\mathbf{x}_{k-1}-\mathbf{x}^*\right)\right\}}$$

end for

$$\text{Normalize weights: } w_j \leftarrow \frac{w_j}{\sum_{l=1}^{N_e} w_l}$$

Resample states at time k using these weights
Update background state $\boldsymbol{\mu}$ and background covariance matrix $\mathbf{B}$ from resampled states
Localize/inflate background covariance B
Set $k \leftarrow k+1$ and repeat

22.5 Sigma Point Kalman Filters (SPKF)

The sigma-point approach is based on deterministic sampling of state distribution to calculate the approximate covariance matrices for the standard Kalman filter equations. There are several versions of the SPKF algorithms that include the unscented Kalman filter (UKF) [201,460], the central difference KF (CDKF), [195,322] and their square root versions, [170,447,448]. In [4], they provide a summary of sigma-point KFs and we shall be summarizing the key points in this section with references to obtain more details of the historic development.

Another interpretation of the SP approach is that it implicitly performs a statistical linearization of the nonlinear model through a weighted statistical linear regression to calculate covariance matrices, [155,248,447,448]. In SPKF the model linearization is done through linear regression between n number of points, referred to as sigma points, drawn from a prior distribution of a random variable rather than through a truncated Taylor series expansion at a single point. Therefore, we start by considering a L-dimensional dynamical system represented by a set of discretized state space equations:

$$\boldsymbol{\theta}_k = f\left(\boldsymbol{\theta}_{k-1}, \mathbf{q}_{k-1}\right), \tag{22.75}$$

$$\boldsymbol{\Psi}_k = \mathbf{h}\left(\boldsymbol{\theta}_k, \mathbf{r}_k\right), \tag{22.76}$$

where $\boldsymbol{\theta}_k$ represents the system state vector at time t_k, $f(\cdot)$ is the nonlinear function of the state, $\mathbf{q}_k$ is the random, assumed to be white, model errors, $\boldsymbol{\Psi}_k$ is the measurement function, and $\mathbf{r}_k$ is the zero mean random measurement noise. In [4] they recast the standard KF optimal state update equation as

$$\widehat{\boldsymbol{\theta}}_k = \widehat{\boldsymbol{\theta}}_k^- + \mathbf{K}_k\left(\boldsymbol{\Psi}_k - \widehat{\boldsymbol{\Psi}}_k^-\right), \tag{22.77}$$

where the negative sign superscripts represent the prior or foretasted states, and $\mathbf{K}_k$ is the standard Kalman gain matrix that we have seen throughout, that is optimally chosen such that it minimizes the weighted scalar sum of the diagonal elements of the error covariance matrix $\mathbf{P}_{\boldsymbol{\theta}_k}^-$. This implies that we can write the Kalman gain and error covariance matrices as

$$\mathbf{K}_k = \mathbf{P}_{\boldsymbol{\theta}_k}^- \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_{\boldsymbol{\theta}_k}^- \mathbf{H} + \mathbf{R}\right)^{-1}, \tag{22.78}$$

$$\mathbf{P}_{\boldsymbol{\theta}_k}^- = \mathbb{E}\left[\left(\boldsymbol{\theta}_k - \widehat{\boldsymbol{\theta}}_k^-\right)\left(\boldsymbol{\theta}_k - \widehat{\boldsymbol{\theta}}_k^-\right)^T\right]. \tag{22.79}$$

The analysis error covariance matrix update, as we know represents the change in forecast error covariance when a measurement is employed, is given by

$$\mathbf{P}_{\boldsymbol{\theta}_k} = \left(\mathbf{I} - \mathbf{K}_k\mathbf{H}\right)\mathbf{P}_{\boldsymbol{\theta}_k}^-. \tag{22.80}$$

The technique employed in SPKF is to reinterpret the standard Kalman gain and covariance update equation in such a way that it needs neither the tangent linear model nor the linearized measurement operator.

The SPKF use a different approach to calculating the Kalman gain matrix compared to the EnKF through the following approach. We start with the $\mathbf{P}_{\boldsymbol{\theta}_k}^- \mathbf{H}^T$ which can be interpreted as the cross-covariance $\mathbf{P}_{\boldsymbol{\theta}_k}\widetilde{\boldsymbol{\Psi}}$ between the state and the observation errors, and the remaining expression in (22.78) can be interpreted as the error covariance $\mathbf{P}_{\widetilde{\boldsymbol{\Psi}}_k}$ of the difference between model and observation. Thus we can write the Kalman gain matrix as

$$\mathbf{K}_k \equiv= \mathbf{P}_{\boldsymbol{\theta}_k, \widetilde{\boldsymbol{\Psi}}_k}\mathbf{P}_{\widetilde{\boldsymbol{\Psi}}_k}^{-1}. \tag{22.81}$$

Here $\widetilde{\boldsymbol{\Psi}}_k$ is defined as the error between the noisy observation $\boldsymbol{\Psi}_k$ and its prediction $\widehat{\boldsymbol{\Psi}}_k^-$ such that $\widetilde{\boldsymbol{\Psi}}_k \equiv \boldsymbol{\Psi}_k - \widetilde{\boldsymbol{\Psi}}_k^-$. Thus we have

$$\mathbf{P}_{\boldsymbol{\theta}_k} = \mathbf{P}_{\boldsymbol{\theta}_k}^- - \mathbf{K}_k\mathbf{P}_{\widetilde{\boldsymbol{\Psi}}_k}\mathbf{K}_k^T. \tag{22.82}$$

See [4] for details of this new form. Therefore unlike standard KF, the SPKF algorithms make use of this new interpretation and avoids the use of Jacobians while retaining consistency and accuracy.

The SPKF makes use of the reformulated error covariance to update (22.81) and (22.82) and chooses the ensembles deterministically in such a way that they can capture the statistical moments of the nonlinear model accurately; in other words, the forecast error covariance in (22.80) is computed using deterministically chosen samples, called sigma points, as such the SPKF algorithm implicitly uses the prior covariance update equation (or the analysis error covariance matrix) to calculate the forecast error covariance. Thus we now consider some approaches to address this calculation.

22.5.1 Sigma-Point Unscented KF (SP-UKF)

The SP-UKF [460] is an SPKF that can capture the statistical properties of the model state through a method known as scaled unscented transformation (SUT), [200]. Unlike the EKF, the SP-UKF uses the true nonlinear model and approximates the state distribution using a set of deterministically chosen states, known as sigma points, using SUT. In SP-UKF the state error covariance matrix is calculated from a set of particular ensembles that are generated by sigma points. In [201] it is shown that for the nonlinear model given by (22.75), the number of sigma points needed to compute precisely the mean and covariance of the model state at time k, is $2L+1$: thus, the *sigma-point state vector* is given by

$$\boldsymbol{\chi}_k \equiv \left[\boldsymbol{\chi}_{k,0} \boldsymbol{\chi}^{+}_{k,i} \boldsymbol{\chi}^{-}_{k,i}\right], \quad i = 1, 2, \ldots, L, \tag{22.83}$$

where $\boldsymbol{\chi}_{k,0}$, $\boldsymbol{\chi}^{+}_{k,i}$ and $\boldsymbol{\chi}^{-}_{k,i}$ are the sigma-point vectors.

The selection scheme for choosing the sigma points is based on the SUT that transforms the model state vector according to the following equations:

$$\boldsymbol{\chi}_{k,0} = \boldsymbol{\theta}_k \text{ with corresponding weight } w_0^{(m)} = \frac{\lambda}{(L+\lambda)}; \tag{22.84}$$

$$\boldsymbol{\chi}^{+}_{k,i} = \boldsymbol{\theta}_k + \left[\sqrt{(L+\lambda)\,\mathbf{P}_{\boldsymbol{\theta}_k}}\right]_i, \quad i = 1, 2, \ldots, L,$$

with corresponding weight

$$w_0^{(c)} = \frac{\lambda}{(L+\lambda)} + \left(1 + \alpha^2 + \beta^2\right); \quad \text{and} \tag{22.85}$$

$$\boldsymbol{\chi}^{-}_{k,i} = \boldsymbol{\theta}_k - \left[\sqrt{(L+\lambda)\,\mathbf{P}_{\boldsymbol{\theta}_k}}\right]_i, \quad i = L+1, L+2, \ldots, 2L,$$

with corresponding weight

$$w_i^{(m)} = w_i^{(c)} = \frac{1}{2\,(L+\lambda)}, \quad \text{where} \quad i = 1, 2, \ldots, 2L, \tag{22.86}$$

where $\left[\sqrt{(L+\lambda)\,\mathbf{P}_{\boldsymbol{\theta}_k}}\right]_i$ is the i-th row (or column) of the weighted matrix square root of the covariance matrix, $\mathbf{P}_{\boldsymbol{\theta}_k} \dot{w}_i^{(m)}$ is the weighting term corresponding to the mean, $w_i^{(c)}$ corresponds to the covariance, and $\lambda = \alpha^2\,(L-\kappa) - L$ is a scaling parameter. The parameter α is set to a small positive value ($0 \leq \alpha \leq 1$) and determines the spread of the sigma points around the mean state $\boldsymbol{\theta}_k$. Another control parameter is κ guarantees the positive semi-definiteness of the covariance matrix and is set to a positive value ($\kappa \geq 0$); β is a non-negative weighting term that can be used to incorporate any prior knowledge of the nature of the state distribution.

In [4] they state that the sigma-point vector is the propagated through the nonlinear model given by

$$\boldsymbol{\chi}_k^{\boldsymbol{\theta}} = f\left(\boldsymbol{\chi}_{k-1}^{\boldsymbol{\theta}}, \boldsymbol{\chi}_{k-1}^{\mathbf{q}}\right), \tag{22.87}$$

$$\mathcal{Y}_k^{\boldsymbol{\theta}} = \mathbf{h}\left(\boldsymbol{\chi}_k^{\boldsymbol{\theta}}, \boldsymbol{\chi}_k^{\mathbf{r}}\right), \tag{22.88}$$

where $\boldsymbol{\chi}_k^{\boldsymbol{\theta}}$ is the forecast sigma-point state vector, $\boldsymbol{\chi}_{k-1}^{\mathbf{q}}$ is the sigma-point vector corresponding to the model error, and $\boldsymbol{\chi}_k^{\mathbf{r}}$ corresponds to the observation error. The approximated mean, covariance an cross covariance for the calculation of the Kalman gain are computed as follows:

$$\widehat{\boldsymbol{\theta}}_k^- \approx \sum_{i=1}^{2L} w_i^{(m)} \boldsymbol{\chi}_{k,i}^{\boldsymbol{\theta}}, \tag{22.89}$$

$$\widehat{\boldsymbol{\Psi}}_k^- \approx \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_{k,i}^{\boldsymbol{\theta}}, \tag{22.90}$$

$$\mathbf{P}_{\boldsymbol{\theta}_k}^- \approx \sum_{i=0}^{2L} w_i^{(c)} \left(\boldsymbol{\chi}_{k,i}^{\boldsymbol{\theta}} - \widehat{\boldsymbol{\theta}}_k^-\right)\left(\boldsymbol{\chi}_{k,i}^{\boldsymbol{\theta}} - \widehat{\boldsymbol{\theta}}_k^-\right)^T, \tag{22.91}$$

$$\mathbf{P}_{\tilde{\boldsymbol{\Psi}}_k} \approx \sum_{i=0}^{2L} w_i^{(c)} \left(\mathcal{Y}_{k,i} - \widehat{\boldsymbol{\Psi}}_k^-\right)\left(\mathcal{Y}_{k,i} - \widetilde{\boldsymbol{\Psi}}_k^-\right)^T, \tag{22.92}$$

$$\mathbf{P}_{\boldsymbol{\theta}_k \tilde{\boldsymbol{\Psi}}_k} \approx \sum_{i=0}^{2L} w_i^{(c)} \left(\boldsymbol{\chi}_{k,i}^{\boldsymbol{\theta}} - \widehat{\boldsymbol{\theta}}_k^-\right)\left(\mathcal{Y}_{k,i} - \widehat{\boldsymbol{\Psi}}_k^-\right)^T. \tag{22.93}$$

Thus the Kalman gain matrix can be calculated using (22.81) and the state covariance is updated using (22.82).

22.5.2 Sigma Point Central Difference KF (SP-CDKF)

In this version of a sigma point KF the analytical derivatives that make up the linearization in the EKF are replaced with central differences, hence the name, and are based upon Stirling's interpolation formula. Thus the nonlinear model form (22.75) can be approximated by

$$f(\boldsymbol{\theta}_k) \approx \overline{f}\left(\overline{\boldsymbol{\varphi}}_k\right) + \widetilde{D}_k + \frac{1}{2}\widetilde{D}_k^2, \tag{22.94}$$

where $\overline{f}\left(\overline{\boldsymbol{\varphi}}_k\right)$ is the linearized model, $\widetilde{D}_k$ and $\widetilde{D}_k^2$ are the central divided difference operators. The linearization here is achieved by using a linear transformation that statistically decouples the state vector, $\boldsymbol{\theta}_k$, and is based upon the square root factorization of the model covariance matrix and is given by

$$\overline{\boldsymbol{\phi}}_k = \mathbf{S}_{\boldsymbol{\theta}_k}^{-1} \overline{\boldsymbol{\theta}}_k, \tag{22.95}$$

$$\widetilde{f}\left(\boldsymbol{\varphi}_k\right) = f\left(\mathbf{S}_{\boldsymbol{\theta}_k}, \overline{\boldsymbol{\varphi}})_k\right) = f\left(\overline{\boldsymbol{\theta}}_k\right). \tag{22.96}$$

Here $\overline{\boldsymbol{\theta}}_k$ is the mean state and $\mathbf{S}_{\boldsymbol{\theta}}$ is the Cholesky factor of the updated covariance matrix (22.82), that satisfies the condition:

$$\mathbf{P}_{\boldsymbol{\theta}_k} = \mathbf{S}_{\boldsymbol{\theta}} \mathbf{S}_{\boldsymbol{\theta}}^T . \tag{22.97}$$

The first and second order central divided difference operators can be written as

$$\widetilde{D}_k = \left(\sum_{i=1}^{L} \left(\boldsymbol{\phi}_k - \overline{\boldsymbol{\phi}}_k \right)_i \mathbf{m}_i \mathbf{d}_i \right) \widetilde{f} \left(\overline{\boldsymbol{\phi}}_k \right), \tag{22.98}$$

$$\widetilde{D}_k^2 = \left(\sum_{i=1}^{L} \left(\boldsymbol{\phi}_k - \overline{\boldsymbol{\phi}}_k \right)_i^2 \mathbf{d}_i^2 + \sum_{j=1}^{L} \sum_{\substack{q=0 \\ q \neq j}}^{L} \left(\boldsymbol{\phi} - \overline{\boldsymbol{\phi}}_k \right)_j \left(\boldsymbol{\phi} - \overline{\boldsymbol{\phi}}_k \right)_q \left(\mathbf{m}_j \mathbf{d}_j \right) \left(\mathbf{m}_q \mathbf{d}_q \right) \right) \widetilde{f} \left(\overline{\boldsymbol{\phi}}_k \right), \tag{22.99}$$

where $\mathbf{m}_i$, $\mathbf{d}_i$ and $\mathbf{d}_i^2$ are the mean, partial first-order, and partial second-order central divided difference operators, respectively, defined as

$$\mathbf{m}_i \widetilde{f} \left(\overline{\boldsymbol{\phi}}_k \right) = \frac{1}{2} \left[f \left(\overline{\boldsymbol{\theta}}_k + \delta \mathbf{s}_{\theta_i} \right) + f \left(\overline{\boldsymbol{\theta}}_k - \delta \mathbf{s}_{\theta_i} \right) \right], \tag{22.100}$$

$$\mathbf{d}_i \widetilde{f} \left(\overline{\boldsymbol{\phi}}_k \right) = \frac{1}{2\delta} \left[f \left(\overline{\boldsymbol{\theta}}_k + \delta \mathbf{s}_{\theta_i} \right) - f \left(\overline{\boldsymbol{\theta}}_k - \delta \mathbf{s}_{\theta_i} \right) \right], \tag{22.101}$$

$$\mathbf{d}_i^2 \widetilde{f} \left(\overline{\boldsymbol{\phi}}_k \right) = \frac{1}{2\delta^2} \left[f \left(\overline{\boldsymbol{\theta}}_k + \delta \mathbf{s}_{\theta_i} \right) + f \left(\overline{\boldsymbol{\theta}}_k - \delta \mathbf{s}_{\theta_i} \right) - 2 f \left(\overline{\boldsymbol{\theta}}_k \right) \right], \tag{22.102}$$

where δ is the central difference step size and $\mathbf{s}_{\theta_i}$ is the i-th column of the Cholesky factor of the covariance updated error covariance matrix (22.82)

$$\mathbf{s}_{\theta_i} = \left(\sqrt{\mathbf{P}_{\boldsymbol{\theta}_k}} \right)_i . \tag{22.103}$$

To implement the SP-CDKF augmented state vectors are constructed by concatenating the original state, model, and observation error vectors. The augmented sigma-point state vectors are calculated using the following selection scheme:

$$\begin{aligned}
&\boldsymbol{\chi}_{k,0} = \overline{\boldsymbol{\theta}}_k \quad \text{with corresponding weights} \quad w_o^{(m)} = \frac{\delta^2 - L}{\delta^2}, \\
&\boldsymbol{\chi}_{k,i}^{+} = \boldsymbol{\theta}_k + \left(\sqrt{\delta^2 \mathbf{P}_{\boldsymbol{\theta}_k}} \right)_i, \quad \text{where} \quad i = 1, 2, \ldots, L, \\
&\qquad \text{with corresponding weight} \quad w_i^{(m)} = \frac{1}{2\delta^2}, \quad \text{where} \quad i = 1, 2, \ldots, 2L; \\
&\boldsymbol{\chi}_{k,i}^{-} = \boldsymbol{\theta}_k - \left(\sqrt{\delta^2 \mathbf{P}_{\boldsymbol{\theta}_k}} \right)_i, \quad \text{where} \quad i = L+1, L+2, \ldots, 2L, \\
&\qquad \text{with corresponding weight} \quad w_i^{(c_1)} = \frac{1}{4\delta^2}, \quad \text{where} \quad i = 1, 2, \ldots, 2L, \quad \text{and} \\
&w_i^{(c_2)} = \frac{\delta^2 - 1}{4\delta^2}, \quad \text{where} \quad i = 1, 2, \ldots, 2L.
\end{aligned} \tag{22.104}$$

The augmented sigma-point vectors are the propagated through the approximated nonlinear model from (22.94), and the approximated mean model state vector can be computed by

$$\widehat{\boldsymbol{\theta}}_k^- = \mathbb{E}\left[\widetilde{f}\left(\overline{\boldsymbol{\phi}}_k\right) + \widetilde{D}_k + \frac{1}{2}\widetilde{D}_k^2\right], \tag{22.105}$$

$$\approx \frac{\delta^2 - L}{\delta^2} f\left(\widehat{\boldsymbol{\theta}}_{k-1}\right) + \frac{1}{2h^2}\sum_{i=1}^{L}\left[f\left(\widehat{\boldsymbol{\theta}}_{k-1} + \delta \mathbf{s}_{\theta_i}\right) + f\left(\widehat{\boldsymbol{\theta}}_{k-1} - \delta \mathbf{s}_{\theta_i}\right)\right], \tag{22.106}$$

$$\approx \sum_{i=0}^{2L} w_i^{(m)} \boldsymbol{\chi}_{k,j}^{\theta}. \tag{22.107}$$

Thus the measurement state, the forecast covariance, and the cross-covariance for the calculation of the Kalman gain are given by

$$\widehat{\boldsymbol{\Psi}}_k^- \approx \sum_{i=0}^{2L} w_i^{(m)} \mathcal{Y}_{k,i}^{\theta}, \tag{22.108}$$

$$\mathbf{P}_{\boldsymbol{\theta}_k}^- \approx \sum_{i=1}^{L}\left[w_i^{(c_1)}\left(\boldsymbol{\chi}_{k,i}^{\theta} - \boldsymbol{\chi}_{k,L+i}^{\theta}\right)^2 + w_i^{(c_2)}\left(\boldsymbol{\chi}_{k,i}^{\theta} + \boldsymbol{\chi}_{k,L+i}^{\theta} - 2\boldsymbol{\chi}_{k,0}^{\theta}\right)^2\right], \tag{22.109}$$

$$\mathbf{P}_{\widetilde{\boldsymbol{\Psi}}_k}^- \approx \sum_{i=1}^{L}\left[w_i^{(c_1)}\left(\mathcal{Y}_{k,i}^{\theta} - \mathcal{Y}_{k,L+i}^{\theta}\right)^2 + w_i^{(c_2)}\left(\mathcal{Y}_{k,i}^{\theta} + \mathcal{Y}_{k,L+i}^{\theta} - 2\mathcal{Y}_{k,0}^{\theta}\right)^2\right], \tag{22.110}$$

$$\mathbf{P}_{\boldsymbol{\theta}_k\widetilde{\boldsymbol{\Psi}}_k} \approx \sum_{i=0}^{L} w_i^{(m)}\left(\boldsymbol{\chi}_{k,i}^{\theta_k} - \widehat{\boldsymbol{\theta}}_k^-\right)\left(\mathcal{Y}_{k,i} - \widehat{\boldsymbol{\Psi}}_k^-\right)^T, \tag{22.111}$$

$$\approx \sqrt{w_1^{(c_1)}\mathbf{P}_{\boldsymbol{\theta}_k}^-}\left(\mathcal{Y}_{k,1:L}^{\theta} - \mathcal{Y}_{k,L+1:2L}^{\theta}\right)^T. \tag{22.112}$$

There has been work recently to help improve the performance of the SPKF reducing the influence of observations on distant state variables by employing a localization scheme to suppress spurious correlations between distant locations in the error covariance matrix, [420].

22.6 Summary

In this chapter we have introduced the theory of MC-based integration, and specifically the application of the MC methods for the inverse problem, which is the solution to the Bayesian problem. We introduced different sampler approaches for the MC methods, but specifically the more widely used Metropolis-Hastings algorithm, which is more commonly known as the MCMC approach. The MCMC algorithm assumes that we only have memory of the state at the previous iteration and not of the earlier values that we have seen of the posterior PDF, which is what the MCMC approach does for the inverse problems.

We have seen that the MCMC methods are used quite extensively in the geosciences to infer information about the different geophysical systems at a specific time, for a given set of observations. We have seen that as a result of the PDFs built by the MCMC method, we can test if the current Gaussian

assumptions are valid for the variational- and ensemble-based data assimilation systems. We have seen in the results from [456] that there are in fact quite strong lognormal and mixed lognormal-Gaussian signals for certain cloud microphysical parameters, which implies that any Gaussian-based assimilation system being applied with these parameters will be suboptimal. The MCMC approach can also be used to quantify the performance of the other data assimilation systems' posterior/analysis distribution and see if they match the MCMC estimates [343].

Next we introduced the particle filters, and show that there had been a lot of progress since the first edition. We also focused on one version of the particle filter called the Local Particle Filter (LPF) that has had some quite good success recently. We also presented a particle smoother that has been developed since the first edition. We finished this chapter by introducing a version of the Kalman filter that is similar to the particle filters called the sigma-point Kalman filters.

We now move on to the first of the two new chapters for this edition: Lagrangian data assimilation.

This page intentionally left blank

CHAPTER

Lagrangian Data Assimilation 23

Contents

When designing the second edition of this textbook, Lagrangian data assimilation had two meaning; the first was with respect to re-defining the data assimilation system in a Lagrangian formulation, rather than Eulerian; the second was to do with assimilating Lagrangian data using different forms of data assimilation algorithms. Originally, this chapter was going to be a small subsection in the variational data assimilation chapter; however, upon further research it became apparent that this work needed its own chapter, as it crosses all the different forms of data assimilation scheme that have been presented. Thus in this chapter we present how many of the data assimilation approaches are adapted to work with Lagrangian data, referred to as LaDA systems. We should note here that the main driver for this approach comes from both ocean and hydrological data assimilation background, where there are some application in the atmosphere through aerosols assimilation.

23.1 Extended Kalman Filter Approach

The summary that we provide here come from [231] where this paper starts with stating the problem, and hence motivation, for Lagrangian based data assimilation as follow; Lagrangian meters, such as weather balloons and ocean drifters and floats, provide a substantial part of atmospheric and oceanic data. These data are used to reconstruct mean large-scale currents, estimate the rate of relative dispersion, and give insight into the formation, movement, and interaction of coherent flow features, such as eddies.

Trajectories of Lagrangian tracers also contain detailed quantitative information about the dynamics of the underlying flow, but this information, at the time of writing of this paper in 2003, were not used for assimilation into flow models. The reason was that most assimilation schemes in oceanography and

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00027-9

meteorology use model variables computed on a fixed grid in space, whereas the Lagrangian observations are distributed non-uniformly over the space and do not give the data directly in terms of model variables.

In [231] they present a method for the assimilation of Lagrangian (drifter) data that follows a sequential approach. Their approach is based on applying the extended Kalman filter, EKF; [198] to the dynamics in the augmented state space that includes drifter coordinates as extra variables, [191]. The augmented state vector $\mathbf{x} \equiv (\mathbf{x}_F . \mathbf{x}_D)^T$ combines the Eulerian part, $\mathbf{x}_F$ describing the state of the flow, and the Lagrangian part $\mathbf{x}_D$—coordinates of the drifters. In this approach advection equations for $\mathbf{x}_D$ are added to the model, where the equations for the evolution of the Eulerian variables are not changed. The augmented error covariance matrix is evolved by the tangent linear model. On the level of second-order statistics it is stated in [231] this carries all necessary information, including the correlations between the errors in flow variables and the errors in drifter positions.

Due to these correlations, the assimilation of the tracer information corrects not only the tracer part of the state vector, $\mathbf{x}_D$, but also the flow variables, $\mathbf{x}_F$. The correction vector, $\mathbf{Kd}$, is a product of the innovation vector, $\mathbf{d}$, that contains the differences between observed drifter coordinates and those predictions by the model, and the Kalman gain matrix $\mathbf{K}$. The latter has dimension $(N + L) \times L$, where N and L are the dimensions of $\mathbf{x}_F$ and $\mathbf{x}_D$ respectively. The first N rows of the matrix $\mathbf{K}$ give the weights with which drifter observations correct the state variables, they are proportional to the part of the error covariance matrix corresponding to correlations between the flow state and drifter positions. **These correlations always appear because drifter motion depends on the underlying flow**.

An advantage of this scheme is that by including drifters into the dynamical model, and tracking them and their correlations with the flow numerically, allows for the ability to extract maximal information about the flow from drifter observations. This is achieved by computing dynamically consistent weights that are given by the gain matrix, $\mathbf{K}$, at each update step, so that the new information from the observations is distributed according to the actual structure of the flow.

We now present the adaptations to the EKF for this approach from [231]. We start be defining the N-dimensional vector $\mathbf{x}(t)$, to represent the state of the flow, which is obtain by a discretization of the model equations, and it comprises of all relevant dynamical variables. The model describing the evolution of the state vector is a deterministic equation

$$\frac{d\mathbf{x}^f}{dt} = \mathcal{M}\left(\mathbf{x}^f, \mathbf{x}_t\right), \tag{23.1}$$

where $\mathcal{M}$ is the corresponding dynamic operator. The superscript here is referring to the forecast and not the true state $\mathbf{x}^t$. It is stated in [231] that a closed system of equations for $\mathbf{x}^t$ is usually unavailable because the state variables interact with the subgrid-scale processes that are not represented by the state vector.

The system in (23.1) may be considered as the best guess for a deterministic equation for $\mathbf{x}_t$, but it is possible to go further if something is known about the statistics of the small-scale processes. Thus the discretized dynamics of $\mathbf{x}^t$ can be written as the stochastic system

$$d\mathbf{x}^t = \mathcal{M}\left(\mathbf{x}^t, \mathbf{x}_t\right) dt + \boldsymbol{\eta}^t(t)\, dt. \tag{23.2}$$

The noise term $\boldsymbol{\eta}^t(t)$ in (23.2) represents the effect of the unresolved small scales, and it assumed it is a zero-mean Gaussian white noise satisfying

$$\mathbb{E}\left[\boldsymbol{\eta}^t\right] = \mathbf{0}, \quad \mathbb{E}\left[\boldsymbol{\eta}^t(t)\left\{\boldsymbol{\eta}^t(t)\right\}^T\right] = \delta\left(t-t'\right)\mathbf{Q}^t(t), \tag{23.3}$$

where $\mathbf{Q}^t(t)$ is referred to as the system noise covariance matrix.

In [231] they use a sequential data assimilation to update the model every time an observation of the true state becomes available. The observations at t_j can be written in terms of $\mathbf{x}_j^t \equiv \mathbf{x}_j^t\left(t_j\right)$ as

$$\mathbf{y}_j^o = \mathbf{h}_j\left(\mathbf{x}_j^t\right) + \boldsymbol{\varepsilon}_j^t, \tag{23.4}$$

where $\boldsymbol{\varepsilon}_j^t$ are random variables representing errors of the observations, that are assumed to be uncorrelated zero-mean Gaussian with an error covariance matrix

$$\mathbb{E}\left[\boldsymbol{\varepsilon}_j^t\left(\boldsymbol{\varepsilon}_j^t\right)^T\right]. \tag{23.5}$$

The dimension of $\mathbf{y}_j^o$ is equal to the number of observations L_j available at t_j. The number and type of observations can vary at each update time.

To combine the observations with the model prediction the extended Kalman filter is used. A key element of the EKF is the tracking of the evolution of the model error (forecast) covariance matrix, [231],

$$\mathbf{P}^f \equiv \mathbb{E}\left[\left(\mathbf{x}^f - \mathbf{x}^t\right)\left(\mathbf{x}^f - \mathbf{x}^t\right)\right], \tag{23.6}$$

using the tangent linear model, gives a closed equation for $\mathbf{P}^f$ as:

$$\frac{d\mathbf{P}^f}{dt} = \mathbf{M}\mathbf{P}^f + \left(\mathbf{M}\mathbf{P}^f\right)^T + \mathbf{Q}(t), \tag{23.7}$$

where $\mathbf{M}$ is the tangent linear model of the nonlinear model, $\mathcal{M}(t)$, evaluated at $\mathbf{x}^f$, and $\mathbf{Q}(t)$ is the best estimation of the system noise covariance matrix $\mathbf{Q}_j^t(t)$.

At each update time t_j a new analysis state, $\mathbf{x}_j^a$, is sought, such that it minimizes the mean-square error

$$\mathrm{tr}\mathbf{P}_j^a = \mathbb{E}\left[\left(\mathbf{x}_j^a - \mathbf{x}_j^t\right)\left(\mathbf{x}_j^a - \mathbf{x}_j^t\right)^T\right]. \tag{23.8}$$

As we saw earlier, the EKF gives a first order approximation to an optimum analysis state using the model error covariance matrix predicted by the linearized equation in (23.7), where the update is given by

$$\mathbf{x}_j^a = \mathbf{x}_j^f + \mathbf{K}_j\mathbf{d}_j, \tag{23.9}$$

where $\mathbf{K}$ is the Kalman gain matrix as defined before, as well as $\mathbf{d}$ being the innovation vector with dimensions here of L_j.

In the study presented in [231], they state that they consider the case when the observations are provided by Lagrangian tracers. Positions of N_D tracers $\mathbf{x}_D$ are observed at t_j. It is stated at this point that

we should note that an observation of tracer positions at one time does not give any information about the flow, it will contain such information only as the tracer position is also known at some previous time t_k, $k < j$. The new positions $\mathbf{x}_D\left(t_j\right)$ depend on $\mathbf{x}_D\left(t_k\right)$ and the flow in $\left[t_k, t_j\right]$.

To extract the information about the flow from the tracer observations the model state space is augmented, so that the new model state vector

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_F \\ \mathbf{x}_D \end{pmatrix}, \tag{23.10}$$

combines the state vector of the flow $\mathbf{x}_F$ and the vector of the tracer coordinates $\mathbf{x}_D$. Here is the part that is different form the theory earlier on the EKF, [231] introduce Tracer advection equations to the model:

$$\begin{aligned} \frac{d\mathbf{x}_F^f}{dt} &= \mathcal{M}_F\left(\mathbf{x}_F^f, t\right), \\ \frac{d\mathbf{x}_D^f}{dt} &= \mathcal{M}_D\left(\mathbf{x}_F^f, \mathbf{x}_D^t, t\right). \end{aligned} \tag{23.11}$$

The first part of (23.11) is the unchanged original model from (23.1), the second part represents discretized tracer advection equations. The now makes the tangent linear model matrix, $\mathbf{M}$, and the forecast error covariance matrix of dimensions $(N + L) \times (N + L)$ that can be written in a block form as

$$\mathbf{M} \equiv \begin{pmatrix} \mathbf{M}_{FF} & \mathbf{0} \\ \mathbf{M}_{DF} & \mathbf{M}_{DD} \end{pmatrix}, \tag{23.12}$$

$$\mathbf{P}^f \equiv \begin{pmatrix} \mathbf{P}_{FF}^f & \mathbf{P}_{FD}^f \\ \mathbf{P}_{DF}^f & \mathbf{P}_{DD}^f \end{pmatrix}. \tag{23.13}$$

In practice the number of available observations at each update time is much less than the number of degrees of freedom of the model, $L \ll N$, which presents a challenge for successful tracking of the flow. It also means that the increase in the dimension of the model, and therefore in the computational cost of dealing with a bigger system, is relatively small.

The observation function $\mathbf{h}$ corresponding to tracer positions is linear, such that

$$\mathbf{h}_j\left(\mathbf{x}_j^t\right) = \mathbf{H}\mathbf{x}_j^t, \quad \mathbf{H} \equiv \begin{pmatrix} \mathbf{0} & \mathbf{I} \end{pmatrix}, \tag{23.14}$$

where $\mathbf{0}$ is a $L \times N$ matrix of zeros, and $\mathbf{I}$ is a $L \times L$ identity matrix. This results in a special form of the Kalman gain matrix:

$$\mathbf{K} = \begin{pmatrix} \mathbf{P}_{FD} \\ \mathbf{P}_{DD} \end{pmatrix} \left(\mathbf{P}_{DD} + \mathbf{R}\right)^{-1}. \tag{23.15}$$

The weights with which drifter observations correct the flow variables $\mathbf{x}_F$ are given by the first N rows of $\mathbf{K}$, where these are proportional to $\mathbf{P}_{FD}$, the correlations between the flow state and the drifter positions. These correlations always appear because drifter paths depend on the flow, $\mathbf{M}_{DF} \neq 0$. It is then stated in [231] that $\mathbf{P}_{FF}^f$, which is the largest part of the error covariance matrix, does not enter

into (23.15). It is required, however, for the prediction of $\mathbf{P}^f_{FD}$, since (23.7) couples all blocks of $\mathbf{P}^f$. Nevertheless, the absence of the direct dependence of the Kalman gain on $\mathbf{P}^f_{FF}$ suggests that some simplified form of the latter could be used.

In [231] they present results using a point vortex system, where point vortex flows are singular solutions to the 2D Euler's equations, and are used to model 2D flows dominated by strong coherent vortices. We shall not go into details about the model, but the reader is referred to [231] for those details. We will, however, present a summary of the conclusions of using this augmented approach that we have presented.

It is stated in [231] that the assimilation of tracer data into two- and four point vortex systems affords successful tracking of the vortices, provided that the observations are sufficiently frequent and accurate. The filter diverges when the noise levels are increased accounting for the nonlinear effects that are neglected by the EKF.

The performance of the assimilation scheme strongly depends on the initial position of the tracer relative to the Lagrangian structures of the flow. In the case of two vortices the boundaries of these structures can be roughly approximated by the separatrices of the streamfunction in the corotating frame. Numerical examples confirm the strong correlation between the efficiency of the assimilation and the tracer position with respect to the separatrices.

In [231] they perform a comparison of the proposed method with an alternative approach that assimilates flow velocity estimated from consecutive tracer positions, and showed that new approach just presented works better because it takes into account the Lagrangian nature of drifter information.

Numerical simulations with four-point vortices demonstrate the feasibility of tracking multi-vortex systems with few tracer observations assimilated by this method. The performance of the assimilation depends not only on the scales of motion and noise levels, but on the character of the underlying deterministic vortex dynamics, that is to say, on the trajectory separation rates in the parts of the phase space visited during the motion. Chaotic vortex initial conditions resulted in the biggest tracking errors, whereas the tracking in case of the symmetric initialization (that restricted the motion to a stable part of the phase space) was the most accurate.

There has been much work on Lagrangian data assimilation techniques, and as we can see if the EKF can be applied to this problem, then it is likely that there is an ensemble based approaches that could tackle this problem. We shall show this later, but first we consider an early approach in the variational formulation t deal with Lagrangian observations for aerosol transport.

23.2 Variational Lagrangian Data Assimilation

Variational approaches have also been used for Lagrangian based data assimilation as well. One of the earliest version for an atmospheric chemistry problems appears in [125]. The approach proposed here is 4D VAR based, where a Lagrangian approach is adopted that allows for the separation of the dynamics and chemistry.

The 4D VAR cost function for this approach is defined by

$$J(\mathbf{x}_0) = \frac{1}{2}(\mathbf{x}_b - \mathbf{x}_0)^T \mathbf{B}^{-1}(\mathbf{x}_b - \mathbf{x}_0) + \frac{1}{2}\sum_{n=0}^{N}(\mathbf{y}_n - \mathbf{s}_n)^T \mathbf{R}_n^{-1}(\mathbf{y}_n - \mathbf{s}_n), \tag{23.16}$$

where $\mathbf{x}_0$ is the vector of initial parcel concentrations, $\mathbf{x}_b$ is an independent estimate of the initial parcel concentrations, and $\mathbf{B}$ here is the covariance matrix of expected errors in $\mathbf{x}_b$, the observation operator $\mathbf{s}_n$ is a linear function of parcel concentrations $\mathbf{H}\mathbf{x}_n$, where $\mathbf{x}_n$ is the vector of concentrations of all species for all parcels at timestep n.

The next step in [125] is the define a functional, J_m in the form

$$J_m = \frac{1}{2}\sum_{n=m}^{N} (\mathbf{y}_n - \mathbf{s}_n)^T \mathbf{R}_n^{-1} (\mathbf{y}_n - \mathbf{s}_n). \tag{23.17}$$

Now if we consider an infinitesimal variation, $\delta\mathbf{x}_0$, in the initial concentrations. At each subsequent step m of the model, there will be corresponding infinitesimal variations, $\delta\mathbf{x}_m$ and δJ_m in the concentrations and in the functional J_m.

The functional, J_m, depends only on concentrations at step m and later. Since these concentrations are uniquely determined by the equations of the model and the concentrations for any step $i \leq m$, then J_m can be regarded as a function of the concentrations at step i only. The gradient of J_m with respect to the concentrations at step i satisfies the equation

$$\delta J_m = \left(\nabla_{\mathbf{x}_i} J_m\right)^T \delta\mathbf{x}_i. \tag{23.18}$$

It is now assumed that $\nabla_{\mathbf{x}_m} J_m$ is known, as such we wish to calculate $\nabla_{\mathbf{x}_{m-1}} J_{m-1}$ and hence by induction, to calculate

$$\nabla_{\mathbf{x}_0} J = \nabla_{\mathbf{x}_0} J_0 + \nabla_{\mathbf{x}_0}\left\{(\mathbf{x}_b - \mathbf{x}_0)^T \mathbf{B}^{-1} (\mathbf{x}_b - \mathbf{x}_0)\right\}, \tag{23.19}$$

where it can be shown that through the nonlinear model being defined as

$$\mathbf{x}_m = \mathcal{M}_{m-1}\left(\mathbf{x}_{m-1}\right), \tag{23.20}$$

then, for infinitesimal variations

$$\delta\mathbf{x}_m = \mathbf{M}_{m-1}\delta\mathbf{x}_{m-1}, \tag{23.21}$$

where $\mathbf{M}_{m-1}$ is the tangent linear model matrix. Through following the arguments for the derivation of full field 4D VAR from earlier we obtain the gradient of the cost function at initial time as we did in the derivation of 4D VAR earlier. The main point here is that in [125] they go on to test this approach with a chemical transport model, but do introduce what is referred to as the influence function, $\gamma_{i,j,m,n}$ and is defined by

$$\gamma_{i,j,m,n} \equiv \frac{\left(\nabla_{\mathbf{x}_m} J_m\right)_i (\mathbf{x}_m)_i}{\left(\nabla_{\mathbf{x}_n} J_n\right)_j (\mathbf{x}_n)_j}. \tag{23.22}$$

The denominator provides a normalization, so that for $m = n$ the value of the influence function is unity for $i = j$. (For $m = n$ and $i \neq j$, the influence function is zero.) The value of the influence function for species i indicates the sensitivity of the fit to the observation to small relative changes in initial species concentrations. A large absolute value of the influence function for species i due to an observation of species j indicates that observations of species j play an important part in determining

the analyzed initial values for species i. [125] show some promising results from this approach, even as far back as the mid-1990s.

In the non-Gaussian chapter we saw different sets of techniques to change the variable or observation of interest into a more Gaussian like variable to analyze. The same is true for the Lagrangian observation, where in the next subsection we look at techniques to convert the Lagrangian data in to Eulerian.

23.2.1 Converting Lagrangian Data to Eulerian to Assimilate

In this section we consider the work of Taillandier et al. from [421] where we start by considering an Eulerian velocity field, $\mathbf{u}$, characterized by mesoscale structures with typical time scale, T_E, and space scale, R_E. From the Lagrangian point of view, the mesoscale flow is characterized by a time scale, T_L, typical of the autocorrelation of Lagrangian particles released in the flow, [421]. Inertial frequency oscillations with time scale T_I are superimposed to the mesoscale field, and they are assumed to be characterized by $T_I \ll T_E$. This assumption said to be realistic for regional and open ocean flows in mid-latitudes, since T_E is typically of the order of one week or more, while $T_I \leq 1d$.

It is now assumed that P Lagrangian floats are released in the flow at the same time, $t = 0$, at different locations, and a such these Lagrangian floats provide information on their position at the same discrete times $m\Delta t$, $m = 0, 1, \ldots M$, over a period $T_M = M\Delta t$. These observations are denoted by $\mathbf{r}^{obs}(m\Delta t)$, where the $2P$-dimensional vector robs is composed of the plane coordinates of the P float positions.

The objective of the work in [421] is to provide an estimate, $\mathbf{u}^{est}$, of the Eulerian velocity field, given the Lagrangian data, $\mathbf{r}^{obs}$, during the period T_M where $T_M \ll T_E$ is assumed, so that $\mathbf{u}^{est}$ can be considered representative of the mesoscale field.

In [421] the generate an estimation problem performed in the variational approach by using the complete data set from the P floats over the time T_M and a model constraint which numerically describes the float advection in the velocity field.

The reconstruction of $\mathbf{u}^{est}$ is carried out in the following way. The M sequences of length Δt are consecutively considered. For each sequence $((m-1)\Delta t, m\Delta t)$, a first guessed velocity field $\mathbf{u}^b$ and its associated prior trajectories $\mathbf{r}^b$ are considered and, through the procedures we are about to show, corrected to fit the model predicted positions $\mathbf{r}(m\Delta t)$ onto their respective observation $\mathbf{r}^{obs}(\Delta t)$. The corresponding velocity increment $\delta\mathbf{u}$, that corrects $\mathbf{u}^b$ on each sequence, is assumed to be indicative of the flow anomalies characterized by structures with typical space scale R. Therefore, $\delta\mathbf{u}$ is approximated as a time-independent Eulerian velocity field, since $T_M \ll T_E$.

We now consider how to go about implementing this approach. In [421] they start by saying that the float advection in the Eulerian velocity field can be described as the solution of the nonlinear ordinary differential equation; $\dfrac{d\mathbf{r}}{dt} = \mathbf{u}(r)$, with initial condition $\mathbf{r}(t_0)$. For a sequence of length Δt, this can be formulated as $\mathbf{r}(m\Delta t) = \mathbf{H}_{NL}(\mathbf{u})$, where the operator $\mathbf{H}_{NL}$ relates the initial float position, $\mathbf{r}(t_0)$, taken at the beginning of the sequence, $t_0 = (m-1)\Delta t$, to the position $\mathbf{r}(m\Delta t)$ at the end of the sequence. Thus this observation operator describes the float advection during the sequence within the flow $\mathbf{u}$.

The numerical version of the float advection equation is now introduced and is discretized over a horizontal mesh grid of spatial resolution, Δs, with a time step, δt, that is assumed for simplicity to be

a fraction of Δt, as

$$\mathbf{r}(t_n) = \mathbf{r}(t_{n-1}) + \delta t \cdot \mathbf{L}_{NL}(\mathbf{r}(t_{n-1})) \cdot \mathbf{u}(t_{n-1}), \quad n = 1, 2, \ldots \frac{\Delta t}{\delta t}, \tag{23.23}$$

where $\mathbf{L}_{NL}$ is the bilinear Lagrange interpolator used for the velocity projection. So the predicted float positions, $\mathbf{r}(m\Delta t)$, are computed by the recurrence given in (23.23), initialized with the observed positions $\mathbf{r}(t_0) = \mathbf{r}^{obs}((m-1)\Delta t)$.

The next step in [421] is to consider time-independent perturbations $\delta\mathbf{u}$ on the first guessed velocity field $\mathbf{u}^b$. The corresponding variation on the prior float positions $\mathbf{r}^b$ is obtained at each time step (t_{n-1}, t_n) by the linearized perturbation equation relative to (23.23);

$$\delta\mathbf{r}(t_n) = \delta\mathbf{r}(t_{n-1}) + \delta t \cdot \mathbf{L}_{NL}\left(\mathbf{r}^b(t_{n-1})\right) \cdot \delta\mathbf{u} + \delta t \cdot \mathbf{L} \cdot \delta\mathbf{r}(t_{n-1}) \cdot \mathbf{u}^b(t_{n-1}), \quad n = 1, 2, \ldots \frac{\Delta t}{\delta t}, \tag{23.24}$$

where the linear operator $\mathbf{L}$ is obtained by deriving $\mathbf{L}_{NL}$ with respect to $\mathbf{r}$ around the prior float positions $\mathbf{r}^b$. Therefore, the variation $\delta\mathbf{r}(m\Delta t)$ of the float positions around their prior estimate at time $m\Delta t$ are computed by the recurrence relationship given in (23.24), assuming unperturbed floats positions at time $(m-1)\Delta t$, that is to say $\delta\mathbf{r}(t_0) = 0$, and can also formulated as $\delta\mathbf{r}(m\Delta t) = \mathbf{H} \cdot \delta\mathbf{u}$, where the operator $\mathbf{H}$ relates the linearized equation of float advection around their prior trajectories. When considering a one-step integration, (23.24) is reduced to

$$\delta\mathbf{r}(m\Delta t)\,\Delta t \cdot \mathbf{L}_{NL}\left(\mathbf{r}^{obs}((m-1)\Delta t)\right) \cdot \delta\mathbf{u}. \tag{23.25}$$

The variation of float positions at time $m\Delta t$ is identified with the Lagrangian velocity increment at the location $\mathbf{r}^{obs}((m-1)\Delta t)$. In this case, the operator, $\mathbf{H}$ attributes weights at each neighboring grid-point value of the Eulerian velocity increment $\delta\mathbf{u}$. This repartition can either be treated by a Gaussian distribution, or directly by the spatial interpolation coefficients defined from each cell containing $\mathbf{r}^{obs}((m-1)\Delta t)$. When using more than one step for the trajectory computation given in (23.24), the velocity increment $\delta\mathbf{u}$ is projected along the first guessed trajectories, at each $\mathbf{r}^b(t_n), n = 1, 2, \ldots, \frac{\Delta t}{\delta t}$.

As an aside here, [421] make the comment that the trajectory sampling δt defines the resolution in space of this projection; it can be related to the accuracy with which $\mathbf{H}$ provides the predicted variations, $\delta\mathbf{r}(m\Delta t)$ from velocity increments $\delta\mathbf{u}$ taken at the resolution Δs. For relatively coarse spatial resolutions Δs compared to float displacements during Δt, the formulation given by (23.25) is sufficient. Instead, when going to regional scales with higher resolution velocity fields, the formulation given by (23.24) is necessary to cover the whole structure described by the float displacements during Δt. As a consequence, the linearized equation of float advection is augmented with a term correcting the spatial interpolation of the first guessed velocity according to the position variations, which increases the accuracy on the prediction of $\delta\mathbf{r}(m\Delta t)$.

The observed float positions at time, $m\Delta t$, are now considered to provide an estimation of the velocity increment, $\delta\mathbf{u}$, according to the first guess velocity field. So the distance to minimize is defined by the corresponding variation of the float positions $\delta\mathbf{r}(m\Delta t) = \mathbf{H} \cdot \delta\mathbf{u}$, as expressed by the cost function

$$J(\delta\mathbf{u}) = \frac{1}{2}\left(\mathbf{H}_{NL}\left(\mathbf{u}^b\right) + \mathbf{H} \cdot \delta\mathbf{u} - \mathbf{r}^{obs}(m\Delta t)\right)^T \cdot \left(\mathbf{H}_{NL}\left(\mathbf{u}^b\right) + \mathbf{H} \cdot \delta\mathbf{u} - \mathbf{r}^{obs}(m\Delta t)\right). \tag{23.26}$$

A feature that is pointed out at this point of [421] is that the components of this model-data misfit are supposed independent, and associated to Gaussian and homogeneous errors, to properly define this

measure in the least square sense. Notice also that background error information are specified as exact optimization constraints which avoids the introduction of a regularization term in the cost function.

As we have seen with other 3D VAR systems we need the gradient of (23.26) to find the optimal velocity increment. Thus the Jacobian is given by

$$\nabla J = \mathbf{H}^T \cdot \left(\mathbf{H}_{NL} \left(\mathbf{u}^b \right) + \mathbf{H} \cdot \delta \mathbf{u} - \mathbf{r}^{obs} \left(m \Delta t \right) \right), \tag{23.27}$$

and ∇J is numerically used to perform the steepest descent step toward the optimal value for $\delta \mathbf{u}$.

To be able to find the minimum of (23.26) we have that for each sequence, $((m-1)\,\Delta t, m\Delta t)$, the first guessed Eulerian velocity field $\mathbf{u}^b$ is iteratively corrected by velocity increments $\delta \mathbf{u}$ following the incremental formulation of variational data assimilation. In order to reconstruct the P float trajectories starting from the observed positions $\mathbf{r}^{obs}\,((m-1)\,\Delta t)$, updates of prior trajectories $\mathbf{r}^b$ are achieved by the nonlinear operator $\mathbf{H}_{NL}$ imposed as a time-dependent model constraint. In between each prior trajectory update, an optimal velocity increment $\delta \mathbf{u}$ is determined in the time-independent approximation as the model constraint is degraded to the linear operator $\mathbf{H}$. The paper goes on to describe a background error covariance decomposition to precondition the problem but there is not background term in (23.26). The reader is referred to [421] for those details. We do present a brief summary of the results form using this approach.

A feature that caused this approach some problems was when $\Delta t > T_L$, with varying P. The results point to a potential problem that could occur, especially, for floats in coherent structures. As the data do not resolve the mesoscale flow curvature, it can happen that data sets induce trajectories that cross each other, which implies that this produces a convergence in the reconstructed velocity field. Again we refer the reader to [421] for more details. We now consider another variational approach where here they are assimilating the Lagrangian observations directly.

23.2.2 Direct Assimilation of Lagrangian Observations

The work that we present here come from [321]. The summary we present here starts at the description of the Lagrangian data, as the previous section in [321] is about the ocean numerical model. As we have seen Lagrangian data are positions of drifting floats. These floats drift between $z_0 - a$ and $z_0 + a$, where z_0 is given by the user and a here is around 25 m. It is assumed that the floats drift at a fixed depth z_0, so that the horizontal plane $z = z_0$. The position of one float at time t in the plane $z = z_0$ are denoted by $\boldsymbol{\xi}\,(t)^T = (\xi_1, \xi_2)\,(t)$, $\boldsymbol{\xi}\,(t)$ is the solution of the differential equation,

$$\begin{cases} \dfrac{d\boldsymbol{\xi}}{dt} = U\,(t, \boldsymbol{\xi}, z_0)\,, \\ \boldsymbol{\xi}\,(0) = \boldsymbol{\xi}_0 \end{cases}, \tag{23.28}$$

where $U = (u, v)$ is the horizontal velocity of the flow and $\boldsymbol{\xi}_0$ is the initial position of the float. Note: The mapping, $U \mapsto \boldsymbol{\xi}$, that links the variables of the model and the Lagrangian observations, is **nonlinear**.

As part of the implementation of this approach, (23.28) needs to be numerically integrated. In [321] it is stated that this is achieved using a leapfrog scheme. This requires the velocity U along the trajectory of the float, recalling that in semi-Lagrangian approaches the departure point may not coincide with a grid point. To achieve this [321] uses the following continuous two-dimensional interpolation

interp$(U,(x,y))$ of the vector field U at the point (x,y):

$$\begin{aligned} x_1 &= \lfloor x \rfloor, & y_1 &= \lfloor y \rfloor, \\ u_1 &= U(x_1, y_1), & u_2 &= U(x_1+1, y_1), \\ u_3 &= U(x_1, y_1+1), & u_4 &= U(x_1+1, y_1+1), \end{aligned}$$

$$\begin{aligned} \text{interp}(U,(x,y)) = u_1 &+ (u_2-u_1)(x-x_1) + (u_3-u_1)(y-y_1) \\ &+ (u_1-u_2-u_3+u_4)(x-x_1)(y-y_1), \end{aligned}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, (x_1, y_1), (x_1+1, y_1), (x_1, y_1+1) and (x_1+1, y_1+1) are the grid points that are the nearest neighbors to (x,y). This function is piecewise affine with respect to x and y, continuous with respect to (x,y), linear with respect to u. Thus it is not differentiable in (x,y) everywhere. More precisely, it is not differentiable at (x,y) if and only if $x = x_1$ or $y = y_1$. It will be a problem to derive the adjoint code. However, it is accurate enough to approximate the solution of (23.28), and it is very costly to use a differentiable interpolation.

If we now consider the algorithm of this part of the work, then we denote by $\boldsymbol{\xi}_k \equiv \left(\xi_{1,k}, \xi_{2,k}\right)$, the horizontal position of the float at time, t_k, U the horizontal velocity of the fluid at time t_k, U_k the velocity at point $\boldsymbol{\xi}_k$, and h the time step of the ocean model. The algorithm step is then schematically

$$\begin{cases} \boldsymbol{\xi}_k = \boldsymbol{\xi}_{k-2} + 2hU_{k-1}, \\ U_k = \text{interp}(U\boldsymbol{\xi}) \end{cases}.$$

In [321] it is stated that the duration between two samplings is the product of N by the time step h of the code ($h = 1200s$), and is referred to as the **timesampling period**. In order to simulate real floats, errors are added to the simulated observations. Origins of errors are multiple: for acoustic floats, inaccuracy can come from acoustic sources (accuracy of their positioning, clock accuracy, bottom topography—acoustic shadow problem, etc), floats (listening period accuracy, complexity of the trajectory, technical problem—temporary 'deafness', etc) or communications quality. For Argo floats errors are due to drift during ascent and descent and also to drift at the surface between ascent/descent and satellite communication. Error amplitude is around 3 to 4 km for acoustic floats and 2 to 6 km for Argo floats.

Now to the data assimilation; without loss of generality, it is assumed that there is only one assimilated sampling: the dataset is the position $\boldsymbol{\xi}(t_1) = (\xi_1(t_1), \xi_1(t_2))$ in the horizontal plane $z = z_0$ of a single float at a single time t_1, and are labeled as the observation as $\mathbf{y}^o = (\xi_1(t_1), \xi_2(t_1))$. Thus the associated cost function here is

$$J(\mathbf{x}) = \frac{1}{2}\left|\left|\mathbf{h}(\mathcal{M}(\mathbf{x})) - \mathbf{y}^o\right|\right|^2 + \frac{\omega}{2}||\mathbf{x} - \mathbf{x}_n||^2_{\mathbf{B}}. \tag{23.29}$$

The background term as introduced earlier is also being used as a regularization term here through the parameter ω. We have standardized the notation from [321] to be consistent with the other chapters. This new approach is implemented into a near operational ocean incremental 4D VAR system. In [321] they provide the details about the adjoint and the numerical approximation that they have to make to it to implement it. We shall just provide a copy of figure 4 from [321] in Fig. 23.1, from their experiment 1, that is to test the sensitivity to the float network parameters; time sampling of the position measurements, number of floats, drifting level, along with coupled impact of number and time sampling. For

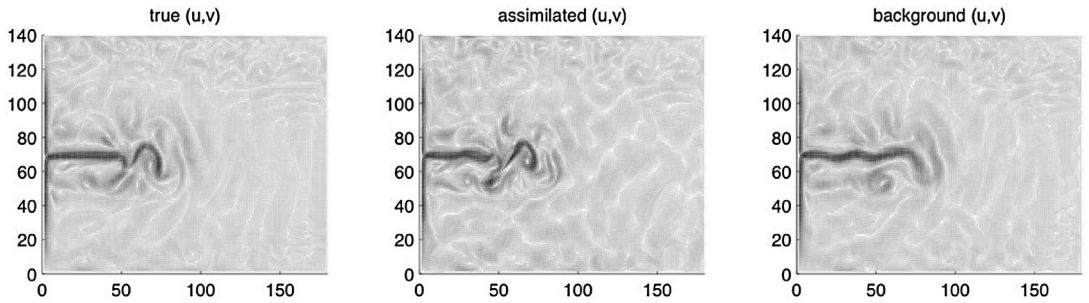


FIGURE 23.1

Copy of figure 4 from [321].

this experiment there are 300 floats drifting at level 4 in the ocean for 10 days, the Lagrangian data are collect once a day.

Fig. 23.1 presents the horizontal velocity field at level 1 at the final time. It contains the true solution, the assimilation solution, as well as the background open loop run. It is clear that the main patterns such as the mid-latitudes jet and bigger eddies are quite similar to the truth.

We move on to the final variational approach in this chapter, which deals with the direct assimilation of Lagrangian data.

23.2.3 Direct Lagrangian Trajectory Variational Assimilation

The work that we present here is from [319] where we shall present what they referred to as the **OceanVar trajectory model**. This model was implemented in the nonlinear observational operator, thus providing a possibility to correct the modeled velocity fields by the observed Argo surfacing coordinates. The forecasted trajectories were calculated from the Eulerian model velocity fields by 5-day integrations of the particle advection equation

$$\frac{d\mathbf{r}}{dt} = \mathbf{u}_L\left(\mathbf{r}\left(t\right), t\right), \tag{23.30}$$

where $\mathbf{r}$ is the float position, and $\mathbf{u}_L$ represents the Lagrangian velocities at the float parking depth during the drift period. The Eulerian velocities $\mathbf{u}$ can be described in the Lagrangian framework as $\mathbf{u}_L\left(\mathbf{r}\left(t\right), t\right) = \mathcal{L}\left(\mathbf{r}\left(t\right)\right)\mathbf{u}\left(t\right)$, where $\mathcal{L}$ is the bilinear Lagrange interpolator. The time-integrated advection equation yields the fully nonlinear trajectory model $H\left(\mathbf{u}\right)$, to be discretized and implemented in the nonlinear observational operator H for OceanVar, and is here presented for one step of integration:

$$\mathbf{r}\left(t_f\right) = \mathbf{r}\left(t_i\right) + \int_{t_i}^{t_f} \mathcal{L}\left(\mathbf{r}\left(t\right)\right)\mathbf{u}\left(t\right)dt, \tag{23.31}$$

where t_i and $t_f = t_i + \Delta t$ indicates the limits of time interval (here, t = 5 days). However, the nonlinearity of this equation imposes a severe analytical problem when the background velocity fields at the observational positions are to be retrieved. Hence, a tangent linear approximation was applied to (23.31), thus yielding the linearized perturbation equation, that will provide the linearized observational operator, H, with the Eulerian velocity increments $\delta\mathbf{u}$:

$$\delta \mathbf{r}\left(t_f\right) = \delta \mathbf{r}\left(t_i\right) + \int_{t_i}^{t_f} \left(\frac{\partial \mathbf{u}_L}{\partial \mathbf{u}} \mid_{r=r_b} \delta \mathbf{u} + \left. \frac{\partial \mathbf{u}_L}{\partial \mathbf{r}} \right|_{u=u_b} \delta \mathbf{r} \right) dt, \tag{23.32}$$

where the position and the Eulerian velocity increments $\delta \mathbf{r} = \mathbf{r} - \mathbf{r}_b$ and $\delta \mathbf{u} = \mathbf{u} - \mathbf{u}_b$ are evaluated around the background velocity $\mathbf{u}_b$ and background position $\mathbf{r}_b$. Moreover, transforming the Lagrangian velocities, $\mathbf{u}_L$ to Eulerian, $\mathbf{u}$, the partial derivatives in (23.32) can be rewritten as $\frac{\partial \mathbf{u}_L}{\partial \mathbf{u}} = \mathcal{L}\left(\mathbf{r}_b\right)$, and $\frac{\partial \mathbf{u}_L}{\partial \mathbf{r}} = \mathbf{L} \cdot \mathbf{u}\left(\mathbf{r}_b\right)$, where $\mathbf{L}$ is the derivative of $\mathcal{L}$ around the background position r_b at the time t_b. The equality of (23.32) is assumed to be fulfilled when the higher-order, nonlinear, terms in (23.31) are negligible.

In the way that this system is implemented, 5-day trajectory predictions are computed for each Argo float from the model velocity fields by the nonlinear trajectory model (23.31), starting at their respective surfacing positions. When a float has fulfilled an "Argo cycle," the OceanVar computes the analyzed float position and the analyzed trajectory, a procedure which requires the present and prior float coordinates. The analyzed position is obtained by minimizing the distance between the present observed float position and the *background position*, which is the last position of the *float* trajectory produced by a 5 days long integration of the trajectory model, in an incremental 3D VAR cost function through the linear operator $\mathbf{H}(\delta \mathbf{u})$. From this analyzed position, the adjoint operator thereafter recalculates the trajectories between the analyzed positions and the prior observed positions.

The OceanVar assimilates data in a daily cycle while trajectories are 5 days long. This inconsistency is neglected by assuming that the innovation is constant throughout the trajectory integration time. That is to say that in (23.32), the background velocity fields are stored during the 5 days long period with the temporal frequency of 6 h, but it is assumed that $\delta \mathbf{u}$, Eulerian, does not change with time. After OceanVar has finished its daily routine, the initial float position for the next Argo cycle is re-set with the observed Argo float position, $\mathbf{r}\left(t_i\right) = \mathbf{r}^{obs}\left(t_i\right)$, the initial float position are held fixed in the OceanVar $\left(\delta \mathbf{r}\left(t_i\right) = 0\right)$, and only the final positions are perturbed by (23.32).

It is stated in [319] that comparisons between the trajectory predictions from numerical experiments (with and without trajectory assimilation) and the observed Argo float positions allows an evaluation of the consistency of the velocity field corrections. The impact of the corrections of the velocity fields can furthermore be assessed quantitatively by calculating the distance between the end points of the trajectories produced at an arbitrary $t = t_i$ and the corresponding observed float positions at $t_f = t_i + \Delta t$. We have a copy of Figure 3 from [319] in Fig. 23.2, that is a schematic of observed and modeled trajectories starting from an observed float position at $t = t_i$.

We now move on to consider the extensions of the EKF approach earlier to the ensemble based Kalman filters to see how they deal with Lagrangian data.

23.3 Lagrangian Ensemble Kalman Filter

The formulation that we present here come from [374], but a lot of the derivation before the ensemble component comes in is from [231]. We begin by recalling the augmented state vector given by

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_F \\ \mathbf{x}_D \end{pmatrix}, \tag{23.33}$$

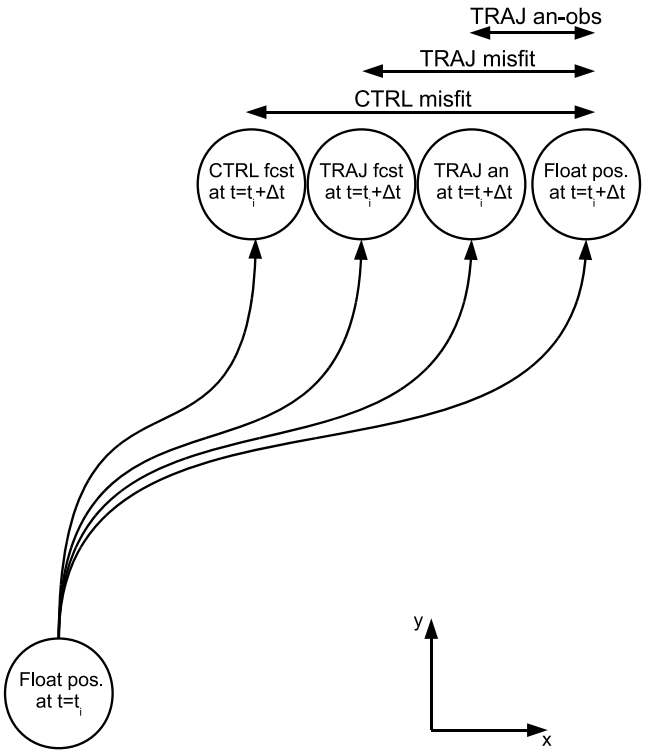


FIGURE 23.2

Copy of figure 3 from [319].

for the flow and drifter variables. We now look at how to calculate $\mathbf{P}^f$ using the EnKF, where we use an ensemble of model forecasts. As always with the ensemble approaches we calculate the ensemble covariance matrix as

$$\mathbf{P}^f \approx \mathbf{P}_e^f = \mathbb{E}\left[\left(\mathbf{x}^f - \overline{\mathbf{x}}^f\right)\left(\mathbf{x}^f - \overline{\mathbf{x}}^f\right)^T\right], \tag{23.34}$$

where the overbar represents the ensemble mean, given by

$$\overline{\mathbf{x}}^f = \frac{1}{N_e}\sum_{i=1}^{N_E}\mathbf{x}_i^f. \tag{23.35}$$

The ensemble covariance matrix is given by

$$\mathbf{P}_e^f = \frac{1}{N_E - 1}\sum_{i=2}^{N_E}\left(\mathbf{x}_i^f - \overline{\mathbf{x}}^f\right)\left(\mathbf{x}_i^f - \overline{\mathbf{x}}^f\right)^T. \tag{23.36}$$

Having defined the error covariance matrix with (23.35) and (23.36) then it possible to construct a first-order approximation to an optimum analysis state through

$$\mathbf{x}_i^a(t_k) = \mathbf{x}_i^f(t_i) + \mathbf{K}(t_k) + \mathbf{d}_i(t_k), \tag{23.37}$$

where i denotes an individual member from our ensemble forecast. The update for each ensemble member can therefore be interpreted as a linear combination of the model forecast, $\mathbf{x}^f$, and the product of the innovation vector

$$\mathbf{d}_i(t_k) = \mathbf{y}^0(t_k) - \mathbf{H}\mathbf{x}_i^f(t_k) + \widetilde{\boldsymbol{\epsilon}_i^f}(t_k), \tag{23.38}$$

where $\widetilde{\boldsymbol{\epsilon}_i^f}(t_k)$ is an additional error introduced to each ensemble member that is required to circumvent the problem of generating an updated ensemble that has a variance that is too low. The associated Kalman gain matrix is

$$\mathbf{K} \equiv \mathbf{P}_e^f \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_e^f \mathbf{H}^T + \mathbf{R}_e\right)^{-1}. \tag{23.39}$$

We recall now that when using this form of the augmented model, the observation operator simplifies to $\mathbf{H} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \end{pmatrix}$. The ensemble analysis error covariance matrix is given by

$$\mathbf{P}_e^a = (\mathbf{I} + \mathbf{K}\mathbf{H})\,\mathbf{P}_e^f, \tag{23.40}$$

which is consistent with the EKF version we presented at the start of this chapter. Recalling also from that first section, the partitioning of the forecast error covariance matrix, where here this has be replaced with the ensemble error covariance matrix which can also be partitioned as

$$\mathbf{P}_e = \begin{pmatrix} (\mathbf{P}_{FF})_e & (\mathbf{P}_{FD})_e \\ (\mathbf{P}_{DF})_e & (\mathbf{P}_{DD})_e \end{pmatrix}, \tag{23.41}$$

and by combining with the block observation operator results in a similar expression as before for the Kalman gain matrix

$$\mathbf{K} = \begin{pmatrix} (\mathbf{P}_{FD})_e \\ (\mathbf{P}_{DD})_e \end{pmatrix} \left((\mathbf{P}_{DD})_e + \mathbf{R}_e^{-1}\right)^{-1}. \tag{23.42}$$

In [374] they indicate that only those elements of $\mathbf{P}_e$ appearing in (23.42) need to be computed in the EnKF. In general $(\mathbf{P}_{FF})_e$ is computationally the most expensive part in constructing the covariance matrix. It is important to emphasize that while the form of $\mathbf{K}$ will be identical in both the EKF and the EnKF formulation, the above simplifications cannot be fully exploited in the EKF.

However, the issue of no having a large enough ensemble does affect this formulation as well, and according to [374], there is a need to bring in a localization to aid the filter. This is achieved through the introduction of a localization matrix $\boldsymbol{\rho}$ such that

$$\mathbf{K} = \begin{pmatrix} \boldsymbol{\rho}_{FD} \circ (\mathbf{P}_{FD})_e \\ \boldsymbol{\rho}_{DD} \circ (\mathbf{P}_{DD})_e \end{pmatrix} \left(\boldsymbol{\rho}_{DD} \circ (\mathbf{P}_{DD})_e + \mathbf{R}_e^{-1}\right)^{-1}. \tag{23.43}$$

Here $\boldsymbol{\rho}_{FD}$ is a $N \times L$ matrix and $\boldsymbol{\rho}_{DD}$ is a $L \times L$ matrix. The operator $\circ$ denotes the Schur product of two matrices. The elements of correspond to a distance-dependent cutoff function. In [374] they

employ a smooth cutoff function and of the form [165]. In Lagrangian data assimilation, both $\boldsymbol{\rho}_{FD}$ and $\boldsymbol{\rho}_{DD}$ are time-dependent matrices since the localization is a function of the changing drifter positions. The localization matrices are, therefore, computed at each assimilation step. In constructing these matrices, we have used the drifter positions at the assimilation step to compute the localization functions although other alternatives may be possible.

In [374] they implement this approach into a shallow water equations model, and performed a vigorous sensitivity of the method, and recommend the reader to this paper. Their conclusions at the end are consistent with other findings when trying to assimilate Lagrangian data in that while they were able to show that this scheme is stable provided the assimilation time interval is on the order of the Lagrangian autocorrelation time scale. However, at larger assimilation time intervals the failure is speculated could be linked to the presence of Lagrangian saddle points in the flow that produce an exponentially rapid separation in the forecast and the true trajectories. This reason has also been speculated on in many of the other papers on this subject.

We now move on to the last of the ensemble methods that is the extension of the ENKF, the LETKF.

23.4 Localized Ensemble Transform Kalman Filter Lagrangian Data Assimilation (LETKF-LaDA)

This approach come from [419], and will again utilizes the flow and drifter augmentation we have seen before. The Local Ensemble Transform Kalman Filter (LETKF) is an ensemble square root filter (EnSRF) as we saw earlier. With respect to localization, there are generally two kinds: in observation space (R-localization), and in model space (B-localization). The LETKF uses R-localization, that selects and weighs local observations in a prescribed region around each grid point while excluding observations outside this region. In [374] it is shown that a proper selection of the localization region is beneficial in using EnKF to assimilate the drifter positions within the shallow water system. To preserve vertically consistent dynamics in each ocean column, no localization is applied in the vertical. As a consequence, surface observations impact the analysis of the entire water column.

This is a result of previous studies showing superior results using this approach to vertical localization when applying LETKF in the ocean, [333,397], only applying variations in the horizontal localization radius is considered. After the localized region is determined, an analysis update to the center grid point at all depths is computed. As we saw earlier, the LETKF assumes Gaussian error statistics that are estimated from the perturbations of the ensemble forecast around the ensemble forecast mean state, where the analysis solution is confined to a maximum $(K-1)$-dimensional linear space defined by the ensemble states, where K is the number of ensemble members.

As a result of applying the localization technique, this allows the global analysis to be formed from a larger dimensional space, though the localized solution is still formed within a linear space limited by the ensemble size. At the analysis step, instead of minimizing the original cost function of the EnKF:

$$J(\mathbf{x}) = \left(\mathbf{x} - \overline{\mathbf{x}}^f\right)^T \left(\mathbf{P}^f\right)^{-1} \left(\mathbf{x} - \overline{\mathbf{x}}^f\right) + \left(\mathbf{y}^o - \mathbf{H}\mathbf{x}\right)^T \mathbf{R}^{-1} \left(\mathbf{y}^o - \mathbf{H}\mathbf{x}\right),$$

with respect to the model state vector x, the LETKF minimizes the equivalent cost function:

$$\widetilde{J}(\mathbf{w}) = (K-1)\,\mathbf{w}^T\mathbf{w} + \left(\mathbf{y}^o - \overline{\mathbf{y}}^f - \mathbf{Y}^f\mathbf{w}\right)^T \mathbf{R}^{-1} \left(\mathbf{y}^o - \overline{\mathbf{y}}^f - \mathbf{Y}^f\mathbf{w}\right), \tag{23.44}$$

where $\mathbf{w}$ is defined as the ensemble weight vector using the formula $\mathbf{x} = \overline{\mathbf{x}}^f + \mathbf{X}^f \mathbf{w}$, within each localization region.

In [419] they designate the LETF to solve the Lagrangian data assimilation problem as LETKF-LaDA for the drifter location case. The steps involved with this approach, where the subscript g represents global, l local, are as follows:

1. Run the dynamical model to obtain the global ensemble forecast states $\mathbf{x}_g^{f(k)} = \begin{pmatrix} \mathbf{x}_{Fg}^{f(k)} & \mathbf{x}_{Dg}^{f(k)} \end{pmatrix}^T$ for $k = 1,2,\ldots,K$, from this obtain the corresponding global ensemble mean, $\overline{\mathbf{x}}_g^f = \begin{pmatrix} \overline{\mathbf{x}}_{Fg}^f & \overline{\mathbf{x}}_{Dg}^f \end{pmatrix}^T$, and the **forecast error perturbation matrix**, $\mathbf{X}_g^f$ whose kth column is $\mathbf{x}_g^{f(k)} - \overline{\mathbf{x}}_g^f$.
2. Using the augmented observation operator for this set up form the $\mathbf{y}_g^{f(k)}$ of the forecast observation vectors by $\mathbf{y}_g^{f(k)} = \mathbf{H}\mathbf{x}_g^{f(k)}$. Compute the corresponding mean $\overline{\mathbf{y}}_g^f$ and error perturbation matrix $\mathbf{Y}_g$ defined in observation space. As the observations are direct observations of the drifters we have, $\mathbf{y}_{Dg}^{f(k)} = \mathbf{x}_{Dg}^{f(k)}$, calculate the mean vectors $\overline{\mathbf{y}}_{Dg}^f = \overline{\mathbf{x}}_{Dg}^f$, and $\mathbf{Y}_D^f = \mathbf{X}_D^f$.
3. Determine the local analysis of the LETKF-LaDA system, using the local arrays of the global terms: $\overline{\mathbf{x}}_l^f$, $\mathbf{X}_l^f$, $\mathbf{Y}_l^f$, $\mathbf{R}_l$, $\mathbf{y}_l^o$ and $\overline{\mathbf{y}}_l^f$. The selection approach of the localization region depends on the type of the analysis variables to be updated. For the fluid variables, [419] have the localization region as a cylinder centered at a horizontal grid point (i, j) with horizontal localization radius γ_{LETKF}, which can be seen in Fig. 1a from [419], that we have a copy of in Fig. 23.3, where γ_{LETKF} is set as a multiple of the baroclinic Rossby radius of deformation. The reader is referred to [419] for more details of the impact of this factor. For global model state variables $\overline{\mathbf{x}}_g^f$ and $\mathbf{X}_g^f$, rows associated to the fluid variables at this grid point (i, j) from all the depth levels are chosen to formulate the corresponding local variables $\overline{\mathbf{x}}_{Fl}^f$ and $\mathbf{X}_{Fl}^f$ In plot (a) in Fig. 23.3, all the observed drifters located in this localized region are marked by their IDs and we define the number of localized observed drifters as N_{Dl}. The rows of $\mathbf{Y}_g^f$, $\mathbf{y}_g^0$, and $\overline{\mathbf{y}}_g^f$ related to this marked drifter IDs are chosen the form

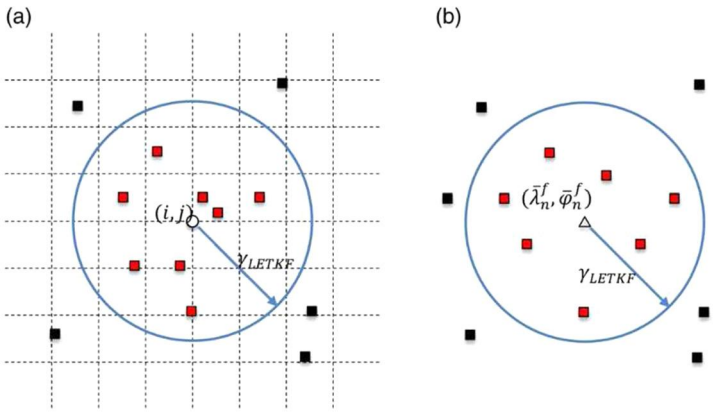


FIGURE 23.3

Copy of Figure 1 from [419].

the local $\mathbf{Y}_l^f$, $\mathbf{y}_l^0$, and $\overline{\mathbf{y}}_l^f$. Plot (b) in Fig. 23.3 illustrates the approach to define the localization region in order to update the local state variables of the simulated drifters, where the selection of the localization region is associated with each drifter ID rather than each model grid point. For each simulated drifter ID n, its forecast ensemble mean location $\left(\overline{\lambda}_n^f, \overline{\varphi}_n^f\right)$ is defined as the center of the corresponding localization region. Localized model state vector $\overline{\mathbf{x}}_{Dl}^f$ and error perturbation matrix $\mathbf{X}_{Dl}^f$ include all the entries associated to this ID n:

$$\overline{\mathbf{x}}_{Dl}^f = \begin{pmatrix} \overline{\lambda}_{Dn}^f \\ \overline{\varphi}_{Dl}^f \end{pmatrix},$$
$$\mathbf{X}_{Dl}^f = \begin{pmatrix} \overline{\lambda}_{Dn}^{f(1)} - \overline{\lambda}_{Dn}^f & \cdots & \overline{\lambda}_{Dn}^{f(K)} - \overline{\lambda}_{Dn}^f \\ \overline{\varphi}_{Dn}^{f(1)} - \overline{\varphi}_{Dn}^f & \cdots & \overline{\varphi}_{Dn}^{f(K)} - \overline{\varphi}_{Dn}^f \end{pmatrix}. \tag{23.45}$$

4. Now follow all the remaining steps from the LETKF as shown earlier, or from [188], or there is a detail description in [419] as well.

There are some promising results shown in [419] and the reader is encourage to read this paper. We now move on to consider a combination approach for the flow and drifter combination.

23.5 Hybrid Particle Filters and Ensemble Kalman Filters Lagrangian Data Assimilation

The approach that we present in this section comes from [396]. As we have seen one initial challenge of assimilating data from Lagrangian instruments is that models of velocity fields are almost always gridded, but the data collected are not on grid points. As stated in [396]; in some ways there is a trade-off between an observation operator that is not local in time and could be nonlinear, which is the case with interpolation, versus the strongly nonlinear dynamics of modeled advected paths, where the paths are observed directly. The latter approach demands an assimilation strategy that can deal with strong nonlinearities.

This motivates the work in [396] where the primary idea behind the proposed hybrid assimilation scheme is: use a particle filter in **low-dimensional, highly nonlinear instrument coordinate variables** and an ensemble Kalman filter in the **high-dimensional flow variables**.

Before we introduce the proposed filter, we start with some notation that is used in [396]. The ensemble from the prior distribution $p(\mathbf{x})$, is denoted $\left\{\mathbf{x}_i^f, w_i^f\right\}_{i=1}^{N_e}$, whereas $\left\{\mathbf{x}_i^a, w_i^f\right\}_{i=1}^{N_e}$ is from the posterior distribution $p(\mathbf{x} \mid \mathbf{y})$.

It is stated in [396] that neither the particle filter nor the ensemble Kalman filter is ideal (either theoretically or practically) in the case of Lagrangian data assimilation. Thus the aim of their hybrid particle–ensemble Kalman filter is to exploit the advantages of each filter by splitting the drifter coordinates away from the flow variables. The high-dimensional, relatively linear Gaussian flow component is estimated via the ensemble Kalman filter, and the low-dimensional, highly nonlinear, and possibly non-Gaussian drifter variables are estimated via a particle filter.

There is a Fokker-Planck equation associated with the flows of the drifters, where in [396] they propose to use a Monte Carlo approximation of the Fokker–Planck equation, by constructing an ensemble of drifter positions, each of which is propagated using a time propagation differential equation. Additionally, instead of applying the particle filter update to the flow variables in each update step, a version of the EnKF is used for weighted ensembles.

In [396] they indicate that there are two main reasons for choosing a combined particle–ensemble Kalman filter strategy; 1) The flow is usually high-dimensional, so a traditional particle filter approximation of the Eulerian variables will require an intractable ensemble size. Since these variables usually do not behave very nonlinearly on time scales of instrument deployment, an EnKF approximation is applied for the updates of these variables. 2) Solving a Fokker–Planck equation for the drifter distribution function can by itself be quite computationally challenging and, in the case of multiple drifters, may not be feasible at all. Hence, a Monte Carlo approximation given by a weighted ensemble of drifters is applied, with the weights updated in a manner similar to a particle filter. Thus, it is expected that the method we are about to present should work well even for realistic models of the ocean flow, augmented by equations for drifter dynamics.

We start by using the same notation as before to represent the flow and drifter variables, but here the authors are assuming a planar fluid flow in which only the position of the drifter on the surface, and not the height of the fluid at its location, is observed. At discrete times t_k, there are observations of the drifter available: $\mathbf{y}^k = \mathbf{x}_D^k + \boldsymbol{\epsilon}^k$, $\boldsymbol{\epsilon}^k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$. At time t_k, the joint distribution of the flow and drifter variables is $p\left(\mathbf{x}_F^k, \mathbf{x}_D^k\right) = p\left(\mathbf{x}_F^k \middle| \mathbf{x}_F^k\right) p\left(\mathbf{x}_F^k\right)$. The marginal distribution on the flow is discretely approximated by $p\left(\mathbf{x}_F^k\right) \approx \frac{1}{N_e} \sum_{i=1}^{N_e} \widetilde{w}_i^k \delta\left(\mathbf{x}_F^k - c_{F,i}^k\right)$ with an ensemble of N_e weighted states $\left\{\mathbf{x}_{F,i}^k, \widetilde{w}_i^k\right\}_{i=1}^{N_e}$. Initially it is mentioned that $\widetilde{w}_i^k = \frac{1}{N_e}$. This then enables the joint probability to be approximated by

$$p\left(\mathbf{x}_F^k, \mathbf{x}_D^k\right) \approx \frac{1}{N_e} \sum_{i=1}^{N_e} p\left(\mathbf{x}_D^k \middle| \mathbf{x}_{F,i}^k\right) \delta\left(\mathbf{x}_F^k - \mathbf{x}_{F,i}^k\right).$$

Next, an approximation of the conditional distribution of the drifters, given each flow ensemble member with a weighted ensemble of M states, is

$$p\left(\mathbf{x}_D^k \middle| \mathbf{x}_{F,i}^k\right) \approx \sum_{j=1}^{N_e} w_{i,j}^k \delta\left(\mathbf{x}_D^k - \mathbf{x}_{D,i,j}^k\right),$$

$\left\{\mathbf{x}_{D,i,j}^k\right\}_{j=1,\ldots,M}$ is the ensemble of drifter states associated with (and subject to) the flow $\mathbf{x}_F^k$ and $\left\{x_{i,j}^k\right\}_{j=1,\ldots,M}$ are the associated weights.

Thus, the full joint distribution is approximated discretely by

$$p\left(\mathbf{x}_F^k, \mathbf{x}_D^k\right) \approx \sum_{i=1}^{N_e} \sum_{j=1}^{M} w_{i,j}^k \delta\left(\mathbf{x}_D^k - \mathbf{x}_{D,i,j}^k\right) \delta\left(\mathbf{x}_F^k - \mathbf{x}_{F,i}^k\right), \tag{23.46}$$

where $\widetilde{w}^k$ is defined in terms of $w_{i,j}^k$ for genera time t_k as

$$\widetilde{w}_i^k = \sum_j w_{i,j}^k, \tag{23.47}$$

so that the weighted ensemble representing the marginal distribution of the flow is given by $\left\{\mathbf{x}_{F,i}^k, \widetilde{w}_i^k\right\}_{i=1}^{N_e}$. Finally, at time t_k we define the following quantities:

$$\widetilde{\mathbf{x}}_{D,i}^k = \frac{1}{\widetilde{w}_i^k} \sum_j \mathbf{x}_{D,i,j}^k w_{i,j}^k, \tag{23.48}$$

$$\overline{\mathbf{x}}_F^k = \sum_i \mathbf{x}_{F,i}^k \widetilde{w}_i^k, \tag{23.49}$$

$$\overline{\mathbf{x}}_D^k = \sum_{i,j} \mathbf{x}_{D,i,j}^k w_i^k = \sum_i \widetilde{\mathbf{x}}_{D,i} \widetilde{w}_i^k. \tag{23.50}$$

That represent the mean of the drifter particles associated with flow member i, the mean of the flow variables, and the mean over all the drifter particles, respectively. See Figure 1 in [396] for a schematic of the set up of the hybrid filter.

After this overview in [396] there is a lot of text which is quite difficult to follow, but is to do with the criteria to resample or not in the particle filter. There is a description of the algorithm that is provided on page 201 which the reader is referred to for the exact details, as well as the mathematical proof of some of the properties in their Appendix.

Before we finish this chapter, we do refer the reader to [11] where this paper provides a detail comparison between particle filters and the ensemble based data assimilation systems to look at the impacts of nonlinearity has on them when applied to Lagrangian data assimilation.

23.6 Summary

This is the first of two new chapters in this edition of the textbook. It was an oversight not to have included it in the first edition, and we hope to have made up for that this time around.

Lagrangian data as we saw, plays an important part in ocean data assimilation, but also in chemical transport in the atmosphere, as well as in flow in river.

We started this chapter with introducing the augmented approach for the extended Kalman filter, before considering different forms of variational approaches to deal with this highly nonlinear observation. We then moved on to the ensemble based approaches before finishing with the combined particle filter ensemble Kalman filter approach.

This is still an active area of research, especially in the ocean data assimilation community, and we recommend all of the papers that we have cited in this chapter to the readers to look up for more extensive details, and to see how the different approaches perform.

We now move on to the second of our new chapters, which is in a field that is gaining much attention in the data assimilation communities, and that is **artificial intelligence and data assimilation**.

This page intentionally left blank

CHAPTER

Artificial Intelligence and Data Assimilation

24

Contents

This is the second of the new chapters for this edition of the textbook, where here we first present helpful definitions along with some of the artificial intelligence (AI) algorithms that have been used/combined with data assimilation algorithms; been used to provide parameters for observation operators that will be assimilated; as well as to detect non-Gaussian signals in the background and observational error and from this determine which version of the 3D VAR full field cost function to minimize. The latter part of this chapter is associated with the application of some of the machine learning (ML) and deep learning (DL) techniques presented in the first three-quarters.

This chapter comprises of 4 sections: Helpful definitions, introduction to ML algorithms, introduction to DL algorithms, and applications of AI with data assimilation. We start, as indicated, with some helpful definitions that are regularly used with AI techniques.

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00028-0

24.1 Helpful Definitions

In this short section we have comprised a series of definitions to consolidate in one place helpful terms that are regularly referred to when dealing with machine and deep learning algorithms.

Definition 24.1. A **training data set** is a data set of examples used during the learning process and is used to fit the parameters. For classification tasks, a supervised learning algorithm looks at the training data set to determine, or learn, the optimal combinations of variables that will generate a good predictive model. The goal is to produce a trained (fitted) model that generalizes well to new, unknown data.

Definition 24.2. A **hyperparameter** is a parameter whose value is used to control the learning process.

Hyperparameters can be classified as model hyperparameters, that cannot be inferred while fitting the machine to the training set because they refer to the model selection task, or algorithm hyperparameters, that in principle have no influence on the performance of the model but affect the speed and quality of the learning process.

Definition 24.3. Classification is the problem of identifying which of a set of categories (subpopulations) an observation (or observations) belongs to. An algorithm that implements classification is known as a **classifier**.

Definition 24.4. Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

Definition 24.5. Kernel density estimation (KDE) is a non-parametric way to estimate the PDF of a random variable. Kernel density estimation is a fundamental data smoothing problem where inferences about the population are made, based on a finite data sample.

The mathematical definition for KDE is let $(x_1, x_2, \ldots, x_n)$ be independent and identically distributed samples drawn from some univariate distribution with an unknown density, f at any given point x. Thus we are interested in estimating the shape of the function f. Its kernel density estimator is given by

$$\widehat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(d\frac{x - x_i}{h}\right), \tag{24.1}$$

where K is the kernel, a non-negative function, and $h > 0$ is a smoothing parameter called the **bandwidth**.

Definition 24.6. The **bandwidth** of the kernel is a free parameter that exhibits a strong influence on the resulting estimate. A optimality criterion used to select this parameter is the expected L_2 risk function, also termed the mean integrated squared error:

$$\text{MISE}(h) = \mathbb{E}\left[\int (\hat{f}_h(x) - f(x))^2\, dx\right]. \tag{24.2}$$

Definition 24.7. Overfitting is the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit to additional data or predict future observations reliably. An overfitted model is a statistical model that contains more parameters than can be justified by the data.

Definition 24.8. Underfitting occurs when a statistical model cannot adequately capture the underlying structure of the data. An under-fitted model is a model where some parameters or terms that would appear in a correctly specified model are missing.

Definition 24.9. A **validation data** set is a data-set of examples used to tune the hyperparameters, sometimes referred to as architecture, of a classifier. In order to avoid overfitting, when any classification parameter needs to be adjusted, it is necessary to have a validation data set in addition to the training and test datasets

Definition 24.10. A **test data set** is a data set that is **independent** of the training data set, but that follows the same probability distribution as the training data set. If a model fit to the training data set also fits the test data set well, minimal overfitting has taken place. A better fitting of the training data set as opposed to the test data set usually points to overfitting.

Definition 24.11. Supervised Learning: These algorithms consist of a *target outcome* variable, sometimes referred to as the dependent variable, that is to be predicted from a given set of **predictors**, or independent variables. From this set of variables, a function is generated that maps the inputs to *desired* outputs. The training process continues until the model achieves a desired level of accuracy on the training data.

Definition 24.12. Unsupervised Learning: In these algorithms there are no targets or outcome variables to predict. These approaches are used for clustering populations in different groups.

Definition 24.13. Reinforcement Learning: The algorithms train the machine to make specific decisions, where the machine is exposed to an environment where it trains itself continually using trail and error. The machine learns from past experiences and attempts to capture best possible knowledge to make accurate decisions.

Given these definitions, we move on to introduce some of the more common machine learning algorithms.

24.2 Introduction to Machine Learning Algorithms

In this section we provide a set of summaries of different machine learning algorithms that are used in the geosciences, as well as possible in conjunction with data assimilation systems.

24.2.1 Linear Regression

The first machine algorithm that we mention we do not have to explain as we saw it in Chapter 17, and that is linear regression. As we saw linear regression is used to estimate real values based upon a continuous variable or variables. Here the relationship between independent and dependent variables is established by fitting the best line, through the equation $y = \beta_0 + \beta_1 x$. There are other forms of linear regression that could be in the form of polynomial regression or curvilinear. There are also multivariate versions referred to as the multiple linear regression where there are multiple independent variables.

If we consider the quadratic polynomial regression, then we are seeking a relationship of the form

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon. \tag{24.3}$$

To able to solve (24.3), we need to solve the following matrix-vector equation

$$\begin{pmatrix} \sum x_i^4 & \sum x_i^3 & \sum x_i^2 \\ \sum x_i^3 & \sum x_i^2 & \sum x_i \\ \sum x_i^2 & \sum x_i & n \end{pmatrix} \begin{pmatrix} \beta_2 \\ \beta_1 \\ \beta_0 \end{pmatrix} = \begin{pmatrix} \sum x_i^2 y_i \\ \sum x_i y_i \\ \sum y_i \end{pmatrix} \tag{24.4}$$

It is also possible to fit a polynomial to degree m as

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_m x^m + \varepsilon, \tag{24.5}$$

as well as for a set of polynomial regression curves as

$$y_i = \beta_{0,i} + \beta_{1,i} x + \beta_{2,i} x^2 + \cdots + \beta_{m,i} x^m + \varepsilon_i, \tag{24.6}$$

where (24.6) can be written in matrix-vector form as

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^m \\ 1 & x_2 & x_2^2 & \cdots & x_2^m \\ 1 & x_3 & x_3^2 & \cdots & x_3^m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^m \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_m \end{pmatrix}, \tag{24.7}$$

which in matrix form is given by

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{24.8}$$

where from using the methods of least squares, the coefficients are approximated by

$$\widehat{\boldsymbol{\beta}} \equiv \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T \mathbf{y}. \tag{24.9}$$

24.2.2 Logistic Regression

We state here that this method is not a regression algorithm but a classification algorithm. Logistic regression is used to model the probability of a certain class or event taking place. Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable. Note, many more complex extensions do exist.

In regression analysis, logistic regression, sometimes referred to as logit regression, is estimating the parameters of a logistic model, which can be seen as a form of binary regression. Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail that is represented by an indicator variable, where the two values are labeled 0 and 1. In the logistic model, the log-odds (the logarithm of the odds) for the value labeled 1 is a linear combination of one or more independent variables, predictors; the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value). The corresponding probability of the value labeled 1 can vary between 0 (certainly the value 0) and 1 (certainly has the value 1), hence the labeling; the function that converts log-odds to probability is the **logistic function**. The unit of measurement for the log-odds scale is called a **logit**, from logistic unit.

In a binary logistic regression model, the dependent variable has two levels or categorical. Outputs with more than two values are modeled by multinomial logistic regression and, if the multiple categories are ordered, by ordinal logistic regression. The logistic regression model models probability of output in terms of input and does not perform statistical classification. Note: It is not a classifier, though it can be used to make a classifier, for instance by choosing a cutoff value and classifying inputs with probability greater than the cutoff as one class, below the cutoff as the other; this is a common way to make a binary classifier.

As we mentioned the model for logistic regression is base upon the logistic function of the form:

$$p(x) = \frac{1}{1 + \exp\left\{-\dfrac{(x-\mu)}{s}\right\}}, \tag{24.10}$$

where μ is a location parameter, (the midpoint of the curve, where $p(\mu) = \frac{1}{2}$, and s is a scale parameter. This expression may be rewritten as:

$$p(x) = \frac{1}{1 + \exp\{-(\beta_0 + \beta_1 x)\}}, \tag{24.11}$$

where $\beta_0 = -\dfrac{\mu}{s}$ and is known as the intercept as it is the vertical intercept, y-intercept, of the line $y = \beta_0 + \beta_1 x$, and $\beta_1 = \dfrac{1}{s}$ is the inverse scale parameter, or rate parameter: these are the y-intercept and slope of the log-odds as a function of x. Conversely, $\mu = -\dfrac{\beta_0}{\beta_1}$ and $s = \dfrac{1}{\beta_1}$.

The usual measure of goodness of fit for a logistic regression uses logistic loss, or log-loss, the negative log-likelihood. For a given x_k and y_k, we have $p_k = p(x_k)$. The p_k are the probabilities that the corresponding y_k will be unity and $1 - p_k$ are the probabilities that they will be zero. Thus we wish to find the values of β_0 and β_1 that give the *best fit* to the data. In the case of linear regression, the sum of the squared deviations of the fit from the data points (y_k), the squared error loss, is taken as a measure of the goodness of fit, and the best fit is obtained when that function is minimized.

Therefore, the log-loss for the k-th point is

$$-\ln p_k \qquad \text{if } y_k = 1, \tag{24.12}$$

$$-\ln(1 - p_k) \qquad \text{if } y_k = 0. \tag{24.13}$$

The log-loss is sometimes interpreted as the *surprisal* of the actual outcome y_k relative to the prediction p_k, and is a measure of *information content*.

When all the data points have their p values we can be combined these into a single expression:

$$-y_k \ln p_k - (1 - y_k) \ln(1 - p_k). \tag{24.14}$$

This expression is more formally known as the **cross entropy** of the predicted distribution $(p_k, (1 - p_k))$ from the actual distribution $(y_k, (1 - y_k))$, as probability distributions on the two-element space of (pass, fail).

The sum of these, the total loss, is the overall negative log-likelihood $-\ell$, and the best fit is obtained for those choices of β_0 and β_1 for which $-\ell$ is minimized.

Alternatively, instead of minimizing the loss, we could maximize the positive log-likelihood:

$$\ell = \sum_{k:y_k=1} \ln(p_k) + \sum_{k:y_k=0} \ln(1-p_k) = \sum_{k=1}^{K} (y_k \ln(p_k) + (1-y_k)\ln(1-p_k)). \tag{24.15}$$

We still have to estimate the parameters but since ℓ is nonlinear with respect to β_0 and β_1, determining their optimum values will require numerical methods. Note that one method of maximizing ℓ is to require the derivatives of ℓ with respect to β_0 and β_1 be zero. That is to say

$$0 = \frac{\partial \ell}{\partial \beta_0} = \sum_{k=1}^{K} (y_k - p_k), \tag{24.16}$$

$$0 = \frac{\partial \ell}{\partial \beta_1} = \sum_{k=1}^{K} (y_k - p_k) x_k. \tag{24.17}$$

This is just an introduction to logistic regression, there are many other forms that are possible for example many explanatory variables, along with the multinomial logistic regression.

24.2.3 Support Vector Machine

Support-vector machine (SVM) is a supervised learning model with associated learning algorithms that analyze data for classification and regression analysis. SVMs are consider one of the most robust prediction methods, being based on statistical learning frameworks. Therefore, given a set of training examples, each marked as belonging to one of two categories, a SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier. SVM maps training examples to points in space so as to maximize the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. We are given a training dataset of n points of the form $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$, where the y_i are either 1 or -1, each indicating the class to which the point $\mathbf{x}_i$ belongs. Each $\mathbf{x}_i$ is a p-dimensional real vector. Thus we would like to find what is referred to as the **maximum-margin hyperplane** that divides the group of points $\mathbf{x}_i$ for which $y_i = 1$ from the group of points for which $y_i = -1$, that is defined so that the distance between the hyperplane and the nearest point $\mathbf{x}_i$ from either group is maximized.

Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying $\mathbf{w}^T\mathbf{x} - b = 0$, where $\mathbf{w}$ is the normal vector to the hyperplane. The parameter $\dfrac{b}{\|\mathbf{w}\|}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

However, there a couple of features of the data that determine the type of hyperplane that we consider. The first is referred to as a **hard-margin**, where if the training data is linearly separable, then it possible to select two parallel hyperplanes that separate the two classes of data, so that the distance between them is as large as possible. The region bounded by these two hyperplanes is called the **margin**, and the maximum-margin hyperplane is the hyperplane that lies halfway between them. With a

normalized or standardized dataset, these hyperplanes can be described by the equations

$$\mathbf{w}^T\mathbf{x} - b = 1, \tag{24.18}$$

where anything on or above this boundary is of one class, with label 1, and

$$\mathbf{w}^T\mathbf{x} - b = -1, \tag{24.19}$$

for anything on or below this boundary is of the other class, with label -1.

Geometrically, the distance between these two hyperplanes is $\dfrac{2}{\|\mathbf{w}\|}$ so to maximize the distance between the planes we want to minimize $\|\mathbf{w}\|$. The distance is computed using the distance from a point to a plane equation. We also have to prevent data points from falling into the margin, and so to prevent this following constraint is added: for each i either

$$\mathbf{w}^T\mathbf{x}_i - b \geq 1 \quad \text{if } y_i = 1,$$

or

$$\mathbf{w}^T\mathbf{x}_i - b \leq -1, \quad \text{if } y_i = -1.$$

These constraints state that each data point must lie on the correct side of the margin, and can be written as

$$y_i\left(\mathbf{w}^T\mathbf{x}_i - b\right) \geq 1, \quad \forall i \quad 1 \leq i \leq n. \tag{24.20}$$

Thus the associated optimization for this description is

$$\text{Minimize } \|\mathbf{w}\| \text{ subject to } y_i(\mathbf{w}^T\mathbf{x}_i - b) \geq 1, \text{ for } i = 1, 2, \ldots, n,$$

where $\mathbf{w}$ and b that solve this problem determine the classifier, $\mathbf{x} \mapsto \text{sgn}\left(\mathbf{w}^T\mathbf{x} - b\right)$ where $\text{sgn}(\cdot)$ is the sign function.

An important consequence of this geometric description is that the max-margin hyperplane is completely determined by the x_i that lie nearest to it. These $\mathbf{x}_i$ are referred to as the **support vectors**.

While it appears to be quite simple if there is a definitive plane that separates the data classifiers it is quite likely when dealing with geophysical data that this is not true. When a nonlinear margin is required then this is referred to at the **soft-margin**.

To be able to extend SVM to cases in which the data are not linearly separable, the following loss function, referred to as the hinge loss function, is useful, where we have

$$\max\left(0, 1 - y_i\left(\mathbf{w}^T\mathbf{x}_i - b\right)\right),$$

where y_i is the i-th target and $\mathbf{w}^T\mathbf{x}_i - b$ is the i-th output.

This function is zero if the constraint in (24.20) is satisfied, that is to say, if $\mathbf{x}_i$ lies on the correct side of the margin. For data on the wrong side of the margin, the function's value is proportional to the distance from the margin. Therefore, the goal of the optimization is to minimize

$$\lambda\|\mathbf{w}\|^2 + \left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i - b)\right)\right], \tag{24.21}$$

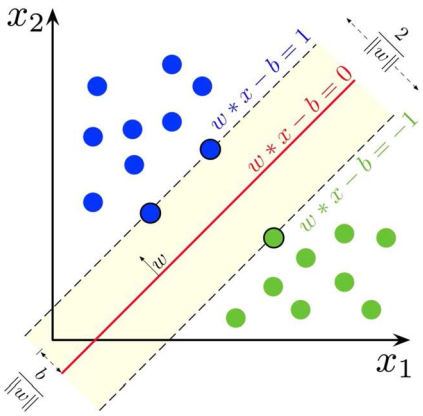


FIGURE 24.1

An illustration of the linear SVM situation.

where the parameter $\lambda > 0$ determines the trade-off between increasing the margin size and ensuring that the $\mathbf{x}_i$ lie on the correct side of the margin. Thus, for sufficiently small values of λ, it will behave similar to the hard-margin SVM, if the input data are linearly classifiable, but will still learn if a classification rule is viable or not.

We have a schematic of a linear situation of the SWM algorithm that has come from https://en.wikipedia.org/wiki/Support-vector_machine#/media/File:SVM_margin.png license by https://creativecommons.org/licenses/by-sa/4.0/m in Fig. 24.1. As we can see for the soft margin approaches, there is a lot more involved that we can present here, but we hope this has induced your interest, as this method, from what we have seen, is held in quite high regard and is seen as a default approach to at least try on your dataset.

24.2.4 Classification and Regression Trees (CART)

Classification and regression trees are different configurations of decision trees, where decision tree learning is a method used in data mining to create a model that predicts the value of a target variable based on several input variables. We start by assuming that all of the input features have finite discrete domains, and there is a single target feature again called the *classification*. Each element of the domain of the classification is called a class.

A decision tree, or classification tree, is a tree in which each internal, non-leaf, node is labeled with an input feature. The arcs coming from a node labeled with an input feature are labeled with each of the possible values of the target feature or the arc leads to a subordinate decision node on a different input feature. Each leaf of the tree is labeled with a class, or a probability distribution over the classes, signifying that the data set has been classified by the tree into either a specific class, or into a particular probability distribution, where if the decision tree is well-constructed, it is skewed towards certain subsets of classes.

A tree is built by splitting the source set, constituting the root node of the tree, into subsets, that constitute the successor children. The splitting is based on a set of splitting rules based on classifica-

tion features. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The recursion is completed when the subset at a node has all the same values of the target variable, or when splitting no longer adds value to the predictions. This process of top-down induction of decision trees is an example of a greedy algorithm, and is the most common strategy for learning decision trees from data.

Returning to our two versions of the decision tree, a **classification tree** is an algorithm where the target variable is fixed, or categorical, and this algorithm is used to identify the class in which a target variable would most likely fall. It could very well be the case that there are multiple values for the categorical dependent variable, and so it does not need to simply be yes/no answers to two decision.

Regression trees refers to an algorithm where the target variable is, and therefore the algorithm is employed to predict it's value. It is built through a process known as binary recursive partitioning, which is an iterative process that splits the data into partitions or branches, and then continues splitting each partition into smaller groups as the method moves up each branch. A regression model is fitted to the target variable using each of the independent variables. After this, the info is split at several points for every experimental variable.

At each such point, the error between the anticipated values and actual values is squared to urge A Sum of Squared Errors (SSE). The SSE is compared across the variables and therefore the variable or point which has rock bottom SSE is chosen because of the split point. This process is sustained recursively.

There are several good textbooks dedicated to CART and we defer the readers there for more details on implementations. This chapter serves to make the reader aware of different algorithms, and for them to determine to progress further with it. However, we do recommend [225] which is a short paper that presents these two approaches in a clear way.

24.2.5 K-Nearest Neighbors

The K-nearest neighbors algorithm, quite often denoted KNN or k-NN is a non-parametric supervised learning method. It is used for classification and regression. In both cases, the input consists of the k closest training examples in a data set. The output depends on whether k-NN is used for classification or regression:

In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor. In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors. k-NN is a type of classification where the function is only approximated locally and all computation is deferred until function evaluation. Since this algorithm relies on distance for classification, if the features represent different physical units or come in vastly different scales then normalizing the training data can improve its accuracy dramatically.

Both for classification and regression, a useful technique can be to assign weights to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor.

The neighbors are taken from a set of objects for which the class (for KNN classification) or the object property value (for KNN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

From a statistical point of view suppose that we have pairs $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$ taking values in $\mathbb{R}^d \times \{1, 2\}$, where Y is the class label of X, so that $X|Y = r \sim P_r$ for $r = 1, 2$ (and probability distributions P_r). Given some norm, $\|\cdot\|$ on $\mathbb{R}^d$ and a point $x \in \mathbb{R}^d$, let $(X_{(1)}, Y_{(1)}), \ldots, (X_{(n)}, Y_{(n)})$ be a reordering of the training data such that $\|X_{(1)} - x\| \leq \cdots \leq \|X_{(n)} - x\|$.

The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the classification phase, k is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label that is most frequent among the k training samples nearest to that query point.

The best choice of k depends upon the data; generally, larger values of k reduces effect of the noise on the classification, but make boundaries between classes less distinct. A good k can be selected by various heuristic techniques. The special case where the class is predicted to be the class of the closest training sample (i.e. when $k = 1$) is called the nearest neighbor algorithm. Note: The accuracy of the k-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the feature scales are not consistent with their importance.

24.2.6 Random Forests

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned.

Before we introduce the process of generating a random forest we introduce the term **bagging**, which is defined as follows: Given a standard training set D of size n, bagging generates m new training sets D_i, each of size n', by sampling from D uniformly and with replacement. By sampling with replacement, some observations may be repeated in each D_i. If $n' = n$, then for large n the set D_i is expected to have the fraction $\left(1 - \frac{1}{e}\right) \approx 63.2\%$ of the unique examples of D, the rest being duplicates. This kind of sample is known as a **bootstrap sample**.

Sampling with replacement ensures each bootstrap is independent from its peers, as it does not depend on previous chosen samples when sampling. Then, m models are fitted using the above m bootstrap samples and combined by averaging the output (for regression) or voting (for classification).

Decision trees are a popular method for various machine learning tasks. Tree learning is seen as being the closest to meeting the requirements for serving as an off-the-shelf procedure for data mining, as it is invariant under scaling and various other transformations of feature values, is robust to inclusion of irrelevant features, and produces inspectable models. However, they are seldom accurate, [166].

In particular, trees that are grown very deep tend to learn highly irregular patterns: they overfit their training sets, indicated by low bias, but very high variance. Random forests are a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance. This comes at the expense of a small increase in the bias and some loss of interpretability, but generally greatly boosts the performance in the final model.

The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set, $X = x_1, x_2, \ldots, x_n$, with responses, $Y = y_1, y_2, \ldots, y_n$, bagging repeatedly, denoted by B times, selects a random sample with replacement of the training set and fits trees to these samples. Thus

For $b = 1, 2, \ldots, B$

1. Sample, with replacement, n training examples from X, Y, denoted by X_b, Y_b.
2. Train a classification or regression tree f_b on X_b, Y_b.

After training, predictions for unseen samples x' can be made by averaging the predictions from all the individual regression trees on x' as

$$\hat{f} = \frac{1}{B} \sum_{b=1}^{B} f_b(x'),$$

or by taking the majority vote in the case of classification trees. We have a schematic of a random forest in Fig. 24.2.

This bootstrapping procedure leads to better model performance because it decreases the variance of the model, without increasing the bias. This implies that while the predictions of a single tree are highly sensitive to noise in its training set, the average of many trees is not, as long as the trees are

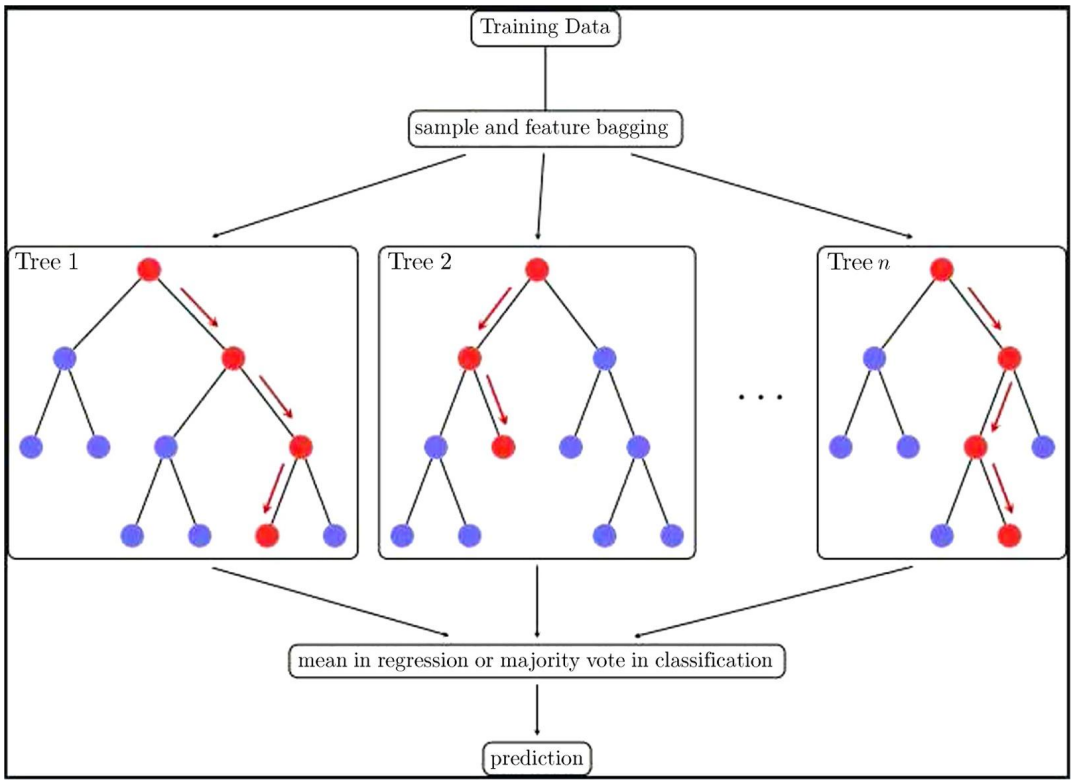


FIGURE 24.2

Illustration of a random forest.

not correlated. Simply training many trees on a single training set would give strongly correlated trees; bootstrap sampling is a way of de-correlating the trees by showing them different training sets.

Additionally, an estimate of the uncertainty of the prediction can be made as the standard deviation of the predictions from all the individual regression trees on x' given by

$$\sigma = \sqrt{\frac{\sum_{b=1}^{B}\left(f_b\left(x'\right) - \hat{f}\right)^2}{B-1}}.$$

The number of samples/trees, B, is a free parameter. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set. An optimal number of trees B can be found using cross-validation, or by observing the out-of-bag error: the mean prediction error on each training sample x_i, using only the trees that did not have x_i in their bootstrap sample. The training and test error tend to level off after some number of trees have been fit.

A relationship between random forests and the K-nearest neighbor algorithm where both approaches can be viewed as weighted neighborhoods schemes. These are models built from a training set $\{(x_i, y_i)\}_{i=1}^{n}$ that make predictions $\hat{y}$ for new points x' by looking at the neighborhood of the point, formalized by a weight function W:

$$\widehat{y} = \sum_{i=1}^{n} W\left(x_i, x'\right) y_i. \tag{24.22}$$

Here, $W\left(x_i, x'\right)$ is the non-negative weight of the i-th training point relative to the new point x' in the same tree. For any particular x', the weights for points x_i must sum to one. Weight functions are given as follows:

- In KNN, the weights are $W\left(x, x'\right) = \dfrac{1}{k}$ if x_i is one of the k points closest to x', and zero otherwise.
- In a tree, $W\left(x_i, x'\right) = \dfrac{1}{k'}$ if x_i is one of the k' points in the same leaf as x', and zero otherwise.

Since a forest averages the predictions of a set of m trees with individual weight functions W_j, its predictions are

$$\widehat{y} = \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{n} W_j\left(x_i, x'\right) y_i = \sum_{i=1}^{n}\left(\frac{1}{m}\sum_{j=1}^{m} W_j\left(x_i, x'\right)\right) y_i.$$

This shows that the whole forest is again a weighted neighborhood scheme, with weights that average those of the individual trees. The neighbors of x' in this interpretation are the points x_i sharing the same leaf in any tree j. In this way, the neighborhood of x' depends in a complex way on the structure of the trees, and thus on the structure of the training set.

This section serves as just a taster for what is available in the machine learning algorithms. Many of the approaches presented here are of the supervised version. We now move on to some more advanced unsupervised approaches collectively referred to as deep learning.

24.3 Introduction to Deep Learning

Before we start we state that while all deep learning algorithms are machine learning algorithms, not all machine learning algorithms are deep learning algorithms, where deep learning is considered an evolution of machine learning. It uses a programmable neural network that enables machines to make accurate decisions **without** human intervention. These algorithms create an **artificial neural network or ANN** that can learn and make intelligent decisions on its own.

A deep learning model is designed to continually analyze data with a logical structure similar to how a human would draw conclusions. To complete this analysis, deep learning applications use a layered structure of algorithms, the ANN just mentioned. The design of an ANN is inspired by the biological network of neurons in the human brain, leading to a learning system that's far more capable than that of standard machine learning models.

While basic machine learning models do become progressively better at performing their specific functions as they take in new data, they still need some human intervention. If a ML algorithm returns an inaccurate prediction, then an engineer has to step in and make adjustments. With a deep learning model, an algorithm can determine whether or not a prediction is accurate through its own neural network—no human help is required. To recap, the key differences between machine learning and deep learning are:

- Machine learning uses algorithms to parse data, learn from that data, and make informed decisions based on what it has learned.
- Deep learning structures algorithms in layers to create an "artificial neural network" that can learn and make intelligent decisions on its own. It is extremely beneficial to data scientists who are tasked with collecting, analyzing and interpreting large amounts of data.
- Deep learning is a subset of machine learning. While both fall under the broad category of artificial intelligence, deep learning is what powers the most human-like AI.

The summary that we present here comes in part from an IEEE Access paper form 2019 titled "Review of Deep Learning Algorithms and Architectures" [394] and well worth a read for more details about deep learning.

24.3.1 Neural Networks (NN)

Neural Network (NN) is a machine learning technique that consists of processing units organized in input, hidden, and output layers. The nodes in each layer are connected to nodes in adjacent layers. Each connection has a weight value. The inputs are multiplied by the respective weights and summed at each unit. The sum then undergoes a transformation based on the activation function. The implementation of a NN consists of the following steps:

1. Acquire training and testing data.
2. Train the network.
3. Make prediction with test data.

According to [394] NN can be classified into the following five types:

1. Feedforward Neural Network,
2. Recurrent Neural Network (RNN),

3. Radial Basis Function Neural Network,
4. Kohonen Self Organizing Neural Network,
5. Modular Neural Network,

In feedforward NN information flows in one direction only; from input to output layer, via hidden nodes. These NN do not form any circles or loopbacks, whereas in RNN the processing units or nodes form a cycle. The output layer becomes the input to the next layer which is typically the only layer in the network, thus the output of the layer become an input to itself.

Radial basis function NN is used in classification, function approximation and time series prediction problems. It consists of input, hidden, and output layers. The hidden layers includes a radial basis function, implemented as a Gaussian function, and each node represents a cluster center. The network learns to designate the inputs to a center and the output layer combines the output of the radial basis function and weight parameters to perform classification or inference.

Kohonen self-organizing NN self organizes the network model into the input data using unsupervised learning. It consists of two fully connected layers, the input and the output layers. The output layer is organized as a two dimensional grid. There is no activation function and the weights represent the attributes of the output layer node. The Euclidean distance between the input data and each output layer node with respect to the wrights are calculated. The weights of the closest node and its neighbors from the input data are updated to bring them close to the input data by

$$w_i\left(t+1\right)=w_i\left(t\right)+\alpha\left(t\right)\eta_{j*i}\left(x\left(t\right)-w_i\left(t\right)\right), \tag{24.23}$$

where $x\left(t\right)$ is the input data at time t, $w\left(i\right)$ is the i-th weight at time t and η_{j*i} is the neighborhood function between the i-th and j-th nodes.

Modular NN breaks down a large network in to smaller independent NN modules. The smaller networks perform specific task that are later combined as part of a single output of the entire network.

Another set of NN that are commonly used are the **Deep Neural Networks (DNN)**, where these some of the ways that they can be implemented:

1. Sparse Autoencoder (SAE),
2. Convolution Neural Networks (CNN),
3. Restricted Boltzmann Machines (RBM).

Autoencoders are NN that learn features or encoding from a given data set in order to perform dimensionality reduction. SAE is a variation of AEs, where some of the units output a value close to zero or are inactive and do not fire. Deep CNN uses multiple layers of unit collections that interact with the input and result in the desired feature extraction. RBM is used to learn probability distribution with the data set.

All of these networks are referred to as backpropagation for training, where this approach uses gradient descent for error reduction by adjusting the weights based on the partial derivative of the error with respect to each weight. NN models can be divided into either a discriminative or generative category.

A discriminative model is a bottom-up approach in which data flows from input layer via the hidden layers to the output layer. They are used in supervised training for classification and regression problems. Whereas, generative models are top-down and data flows in the opposite direction. They

are used in unsupervised pre-training and probabilistic distribution problems. If the input x and corresponding label y are given, a discriminative model learns the probability distribution $p(x|y)$, whereas a generative model learns the joint probability of $p(x, y)$, from which $P(y|x)$ can be predicted.

The training of the NN can be broadly categorized into the following three types:

1. Supervised,
2. Unsupervised,
3. Semi-supervised.

In [394] they state that supervised learning consists of labeled data which is used to train the network, whereas unsupervised learning there is no labeled data set, thus no learning based on feedback. In unsupervised learning, neural networks are pre-trained using generating models such as RBMs and later could be fine-tuned using standard supervised learning algorithms. It is then used on test data set to determine patterns or classifications.

Returning to the three DNN methods we have a brief outline of each one.

Convolution Neural Networks

According to [394] the CNN is based upon the human visual cortex. A CNN consists of a series of convolution and sub-sampling layers followed by a fully connected layer and a normalizing layer. We have an illustration of a CNN in Fig. 24.3 that comes from the following website that has been drawn in TikZ software, https://davidstutz.de/illustrating-convolutional-neural-networks-in-latex-with-tikz/ and we give credit to this site for this figure.

The series of multiple convolution layers perform progressively more refined feature extraction at every layer moving from input to output layers. Fully connected layers that perform classification follow the convolution layers. Sub-sampling or pooling layers are often inserted between each convolution layers. CNN's takes a 2D $\mathbf{n} \times \mathbf{n}$ pixelated image as an input. Each layer consists of groups of 2D neurons called filters or kernels. Unlike other neural networks, neurons in each feature extraction layers of CNN are not connected to all neurons in the adjacent layers. Instead, they are only connected to the

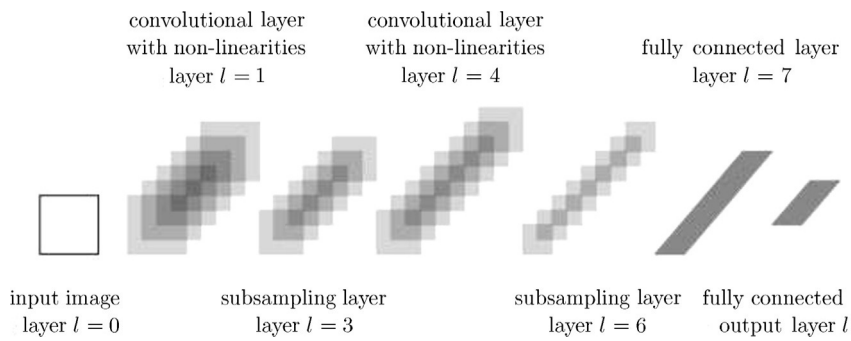


FIGURE 24.3

Illustration of a simple convolution neural network indicating convolutional layers, pooling layers and fully-connected layers without details (number of channels or neurons per layer or the input image size).

spatially mapped fixed sized and partially overlapping neurons in the previous layer's input image or feature map. This region in the input is called local receptive field.

The lowered number of connections reduces training time and chances of overfitting. All neurons in a filter are connected to the same number of neurons in the previous input layer (or feature map) and are constrained to have the same sequence of weights and biases. These factors speed up the learning and reduces the memory requirements for the network. Thus, each neuron in a specific filter looks for the same pattern but in different parts of the input image. Sub-sampling layers reduce the size of the network. In addition, along with local receptive fields and shared weights (within the same filter), it effectively reduces the network's susceptibility of shifts, scale and distortions of images. Max/mean pooling or local averaging filters are used often to achieve sub-sampling. The final layers of CNN are responsible for the actual classifications, where neurons between the layers are fully connected.

Deep CNN can be implemented with multiple series of weight-sharing convolution layers and sub-sampling layers. The deep nature of the CNN results in high quality representations while maintaining locality, reduced parameters and invariance to minor variations in the input image.

There is a very good mathematical and algorithmic description of CNNs in [394], and the reader is referred there for these details.

AutoEncoder (AE)

Autoencoder is a neural network that uses unsupervised algorithm and learns the representation in the input data set for **dimensionality reduction** and to recreate the original data set. The learning algorithm is based on the implementation of the backpropagation.

AEs extend the idea of principal component analysis (PCA), where PCA transforms multi-dimensional data into a linear representation. Whereas AEs can produce nonlinear representation. PCA determines a set of linear variables in the directions with largest variance. The p dimensional input data points are represented as m orthogonal directions, such that $m \leq p$ and constitutes a lower dimensional space. The original data points are projected into the principal directions, thus omitting information in the corresponding orthogonal directions.

PCA focuses on the variances, rather than covariances and correlations, and it looks for the linear function with the most variance. The goal is to determine the direction with the least mean square error, that would then have the least reconstruction error.

AEs use encoder and decoder blocks of non-linear hidden layers to generalize PCA to perform dimensionality reduction and eventual reconstruction of the original data. It uses greedy layer by layer unsupervised pre-training and fin-tuning with backpropagation, [394]. Despite using backpropagation, which is mostly used in supervised training, AEs are considered unsupervised DNN because they regenerate the input $x^{(i)}$ itself instead of a different set of target values $y^{(i)}$, that is to say $y^{(i)} = x^{(i)}$. We have a diagram of an AE in Fig. 24.4.

24.3.2 Restricted Boltzmann Machine (RBM)

Restricted Boltzmann Machine is an ANN where it is possible to apply unsupervised learning algorithm to build non-linear generative models from unlabeled data. We wish to train the network to increase a function (e.g., product or log) of the probability of vector in the visible units so it can probabilistically reconstruct the input. It learns the probability distribution over its inputs. This network comprises of a visible and a hidden layer only. Each unit in the visible layer is connected to all units in the hidden layer and there are no connections between the units in the same layer.

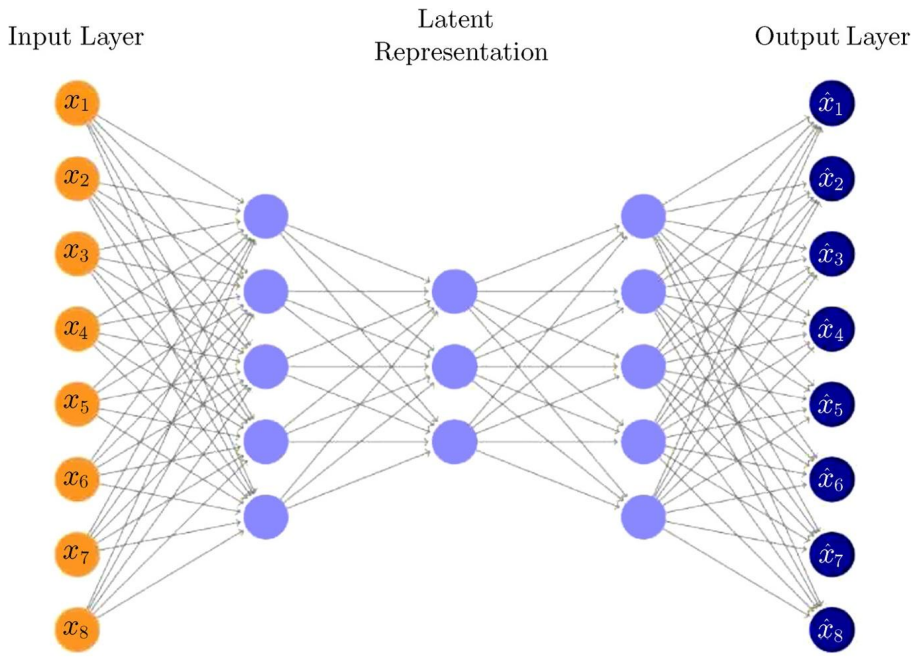


FIGURE 24.4

Illustration of the properties of an AutoEncoder.

The energy E function of the configuration of the visible and hidden units, (v, h) is expressed as

$$E(v,h) = -\sum_{i \varepsilon visible} a_i v_i - \sum_{j \varepsilon hidden} b_j h_j - \sum_{i,j} v_i h_j w_{ij}, \tag{24.24}$$

where v_i and h_j are the vector states of the visible unit i and hidden unit j. a_i and b_j represents the bias of visible and hidden units. w_{ij} denotes the weight between the respective visible and hidden units.

A partition function, Z, represented the sum of all possible pairs of visible and hidden vectors

$$Z = \sum_{v,h} \exp\{E(v,h)\}.$$

The probability of every pair of visible and hidden vectors is given by

$$p(v,h) = \frac{1}{Z} \exp\{-E(v,h)\}, \tag{24.25}$$

where the probability of a particular visible layer vector is given by

$$p(v) = \frac{1}{Z} \sum_{h} \exp\{-E(v,h)\}.$$

When the machine learning algorithm has been selected for the problem at hand, the next part is to decide on the training algorithm.

24.3.3 Training Algorithms

In [394] at this point they make the clarifying statement

> The learning algorithm constitutes the main part of Deep Learning. The number of layers differentiates the deep neural networks from shallow ones. The higher the number of layers, the deeper it becomes. Each layer can be specialized to detect a specific aspect of a feature".

From [394] they say that in [314] in the case of image recognition, the first layer can detect edges and the second can detect higher features, and the third can go up further the complexity order. Even though each layer might learn or detect a define feature, the sequence is not always designed for it, especially in unsupervised learning. Thus the goal of the learning algorithm is to find the optimal values for the weight vectors to solve a class of problem in a domain. In [394] they indicate five training algorithms:

1. Gradient Descent
2. Stochastic Gradient Descent
3. Momentum
4. Lavenberg-Marquardt algorithm
5. Backpropagation through time

All five of these training algorithms are presented in [394] and the reader is referred to that paper for the exact details.

Shortcomings of training algorithms

In [394] they present six shortcomings for the training data presented below:

1. **Vanishing and exploding gradients:** DNN are prone to vanishing, or exploding, gradients due to the inherent way in which gradients are computed layer by layer in a cascading manner with each layer contributing to exponentially decreasing or increasing derivatives. Weights are increased or decreased based on gradients to reduce the cost function or error. Very small gradients can cause the network to take a long time to train, whereas large gradients can cause the training to overshoot and diverge. This is made worse by the non-linear activation functions like sigmoid and tanh functions that squash the outputs to a small range. Since change in weight have nominal effect on the output training could take much longer. This problem can be mitigated using proper weight initialization.
2. **Local Minima:** Local minima is always the global minima in a convex function, that makes gradient descent based optimization fool proof. Whereas in nonconvex functions, backpropagation based gradient descent is particularly vulnerable to the issue of premature convergence into the local minima. A local minima can easily be mistaken for global absolute minima, unless the Hessian is being evaluated to ensure that it is one sign.
3. **Flat Regions:** Flat regions or saddle points also pose similar challenge for gradient descent based optimization in nonconvex high-dimensional functions. The training algorithm could potentially be mislead by this area as the gradient comes to a halt at this point.

4. **Steep Edges:** Steep edges are another section of the optimization surface area where the steep gradient could cause the gradient descent-based weight updates to overshoot and miss a potential global minima.
5. **Training Time:** Training time is an important factor to gauge the efficiency of an algorithm. Most models require exorbitant amount of time and large datasets to train. Often times many of the samples from the datasets do not add value to the training process and in some cases, they introduce noise and adversely affect the training.
6. **Overfitting:** As more neurons are added to DNN, it can model the network for more complex problems. DNN can lend itself to high conformability to training data. But there is also a high risk of overfitting to the outliers and noise in the training data. This can result in delayed training and testing times and result in the lower quality prediction on the actual test data. E.g., in classification or cluster problems, overfitting can create a high order polynomial output that separates the decision boundary for the training set, which will take longer and result in degraded results for most test data set. One way to overcome overfitting is to choose the number of neurons in the hidden layer wisely to match the problem size and type.

[394] also present algorithms to optimize the training algorithms. We do not go into detail here but do the list them here for reference

- Parameter initialization techniques
- Hyperparameter optimization
- Adaptive learning rates
- Batch normalization
- Supervised pretraining
- Dropout.

As we mentioned earlier, [394] is a very good review of deep learning algorithms and we recommend it to anyone looking for a high up overview of deep learning.

However, before we move on to the applications there is one more NN that we have found that has some importance with DA that is not referred to in [394]. These are the **Bayesian Neural Networks, BNN** and are defined as follows:

Definition 24.14. Bayesian neural network (BNN) combines neural network with Bayesian inference, where in BNN, we treat the weights and outputs as the variables and so we are finding their marginal distributions that best fit the data. The ultimate goal of BNN is to quantify the uncertainty introduced by the models in terms of outputs and weights so as to explain the trustworthiness of the prediction.

Below are three key points that we have found from our research on these approaches that differentiate them from the standard NN (SNN).

1. GOAL — SNN focuses on optimization, while BNN focuses on marginalization. Optimization would find one optimal point to represent a weight, while marginalization would treat each weight as a variable and find its distribution.
2. ESTIMATE — The estimate of the parameters for SNN would be maximum likelihood estimators (MLE), while for BNN, the estimate would be the maximum a posteriori (MAP) state or predictive distribution.

3. METHOD — SNN would use differentiation to find the optimal value such as gradient descent. In BNN, since the integrals are hard to determine, we would rely on MCMC, Variational Inference, and Normalizing Flows these kinds of techniques.

If Fig. 24.5 we have a schematic of the difference between a SNN and a BNN, where the lines on the BNN represent the PDFs between those layers and nodes.

24.4 Applications of Artificial Intelligence With Data Assimilation

In this section we shall present some recent application of AI algorithms, be it machine learning, or deep learning, that have helped to improve the performance of the data assimilation systems. The first application we present is work that we have been involved with; detecting non-Gaussian signals in the background fields, in order to determine which version of the variational cost function, Gaussian fits all, or mixed Gaussian-lognormal, although work switching to Gaussian-reverse-lognormal is in progress, to use.

24.4.1 Detection of Non-Gaussian Signals

When visiting the Naval Research Laboratory in the summer of 2011, a question was posed to us about the new lognormal based variational approach in the form of; How would we know when to switch? It took nearly 10 years to be able to develop approaches that appear to enhance the performance of the variational data assimilation schemes and answer this question. The initial work looked at building histograms to indicate from a climatological stand point that through the summer the moisture field was Gaussian but in the winter it was more lognormal (drier) and that it transition through reverse lognormal in the spring. These results can be found in [220]. Recalling the work of [142] in their figure 8 (Fig. 24.6) they present the GPS station LHAS in Lhasa, Tibet that shows a clear bimodal distribution of zenith neutral delay (ZND), which is a measure of the propagation delay imposed by the neutral atmosphere on GPS (radio) signals reaching that station. In this case the rapid drop to zero probability on the upper end of the histogram suggests that this is most likely comprises a lower lognormal and an upper reverse-lognormal mode. The modes themselves are clearly related to the monsoonal seasonality at this site, [142].

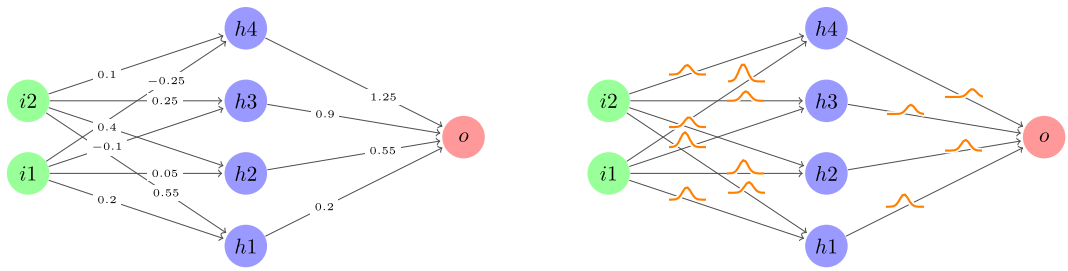


FIGURE 24.5

Schematic of the Standard Neural Network (left), vs the Bayesian Neural Network (right).

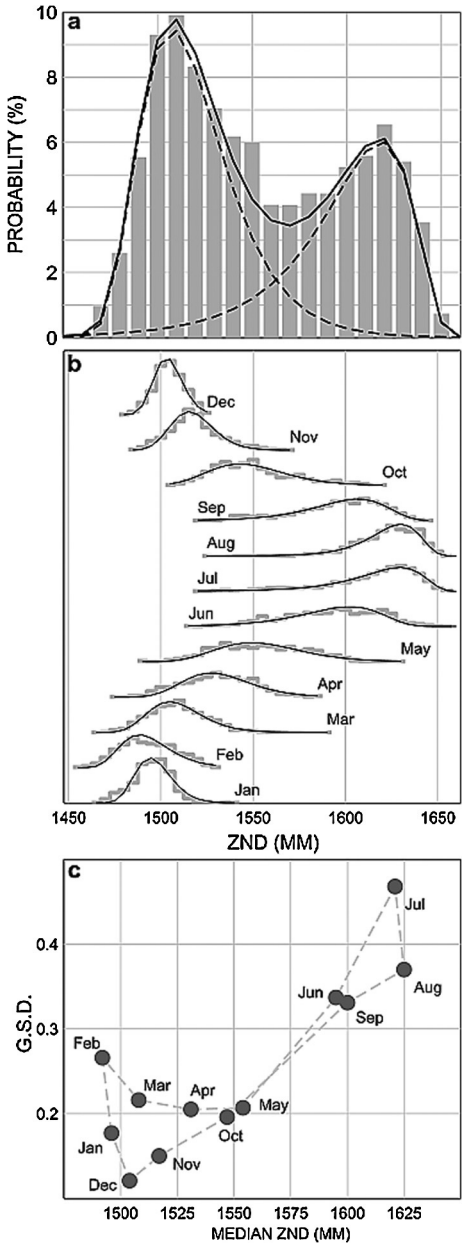


FIGURE 24.6

Copy of figure 8 from [142].

In Chapter 21 we presented histograms of the climatology of the z component of the Lorenz 63 model, where we can see the two modes, along with a dip in the middle, as we can see in Fig. 24.6. Therefore, the question becomes can we separate them out?

In [158] a first attempt was a manual separating out of the attractor through certain inequalities and then fitting the best distribution to that data. We have a copy of figure 5 from [158] in Fig. 24.7 that shows 16 subsets of the z component where we can see that the distribution does change depending on where you are on the attractor. Another important feature to notice is that the data has now almost become unimodal in each of these regions making variational DA more optimal.

However, we cannot manually watch the z component to determine which of these sectors we are in, so two AI techniques were applied to see if the machine could detect the change in the distribution and identify which distribution it was. The two techniques were a SVM with radial basis function kernel, and a NN with 5 layers, with a hyperbolic tangent as the activation function, with a stochastic gradient based optimizer. The two systems were trained with 50,000 times steps and then verified again a further 50,000 time steps. The two systems had two identifiers to determine if the z component had changed from a Gaussian to a lognormal, or vice-versa. Note that we are not determining the reverse lognormal but that it was through this work we became aware of it; these were: 1) difference between the mode and mean was not zero implied that this was lognormal if this difference was positive but Gaussian

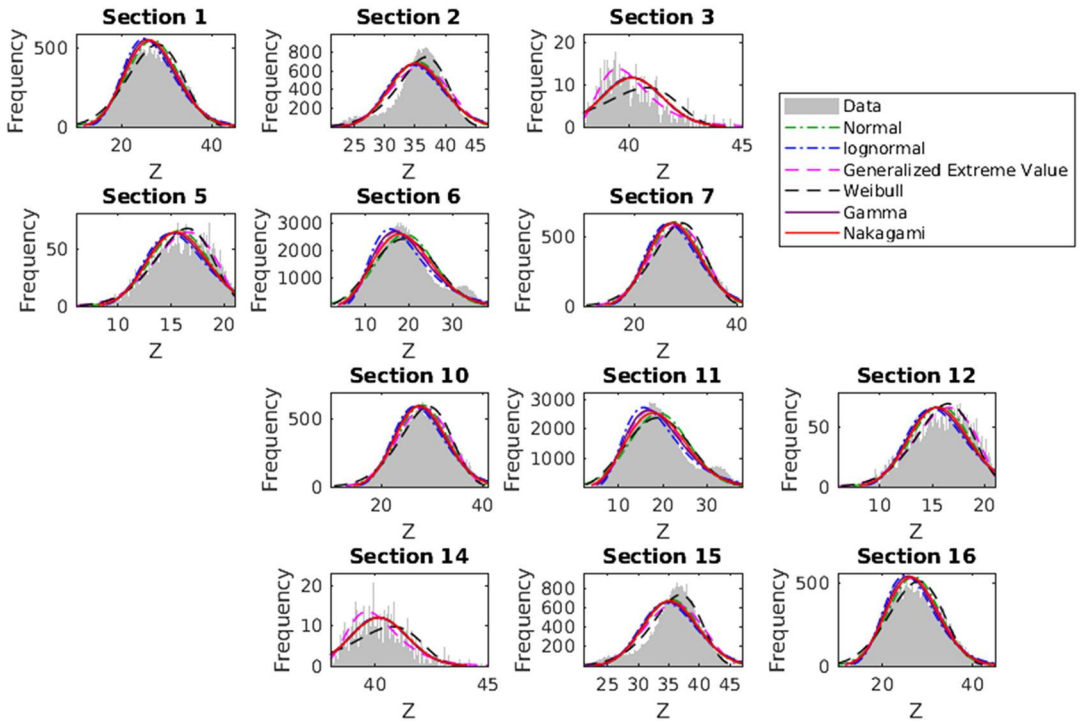


FIGURE 24.7

Copy of figure 5 from [158].

if zero or negative; 2) skewness, zero if it is Gaussian (as well as negative), and lognormal if this is positive.

Both approaches showed great promise in correctly detecting the changes between the distribution, with the SVM appearing to be more consistent and accurate then the NN here. Given this motivation the SVM was implemented into the full field mixed Gaussian-lognormal variational and Gaussian-fits-all scheme with the Lorenz 63 model and results are shown in [159] where for small detection windows, the Gaussian fits all is a good approximation, but when the windows become larger, using the lognormal and Gaussian at difference times did improve the analysis and forecast errors.

24.4.2 Deep Data Assimilation

In [14] a new version of data assimilation is proposed where a deep learning (NN) algorithm is integrated with a data assimilation system. As we have seen data assimilation methodologies improve the levels of confidence in computational prediction, improve numerical forecasted results, by incorporating observational data into a prediction model. However, the error propagation into the forecasting model is not improved by DA, so that, at each step, correction have to be based from scratch without learning from previous experience of error correction.

The strongly nonlinear character of many physical processes of interest can result in the dramatic amplification of even small uncertainties in the input, so that they produce large uncertainties in the system behavior. Because this instability, as many observations are assimilated as possible to the point where a strong requirement to data assimilation is to enable real-time utilization of data to improve predictions. Therefore, the motivation in [14] is to use machine learning method in order to learn a function that accumulates the process of previous assimilation process. As such a NN is used to model this process. The key concept is in recording each step of state correction during an assimilation period and then learn a function in order to capture this updating mechanism. The system model is then revised by composing this learned updating function with the current system model. Such a process continues by further learning the assimilation process with the updated system model. The resulting NN-forecasting model is then a forecasting model with an intrinsic assimilation process. We have a copy of figure 4 from [14] in Fig. 24.8 that shows the schematic of the DDA algorithm. See [14] for the full description of the algorithm as well as proof of properties of the new approach.

24.4.3 Latent Space Data Assimilation by Using Deep Learning

In [334] they state that by capitalizing on the ability of neural network techniques to approximate the solution of partial differential equations, they incorporate deep learning methods into a data assimilation framework. They exploit the latent structure provided by autoencoders to design an ensemble transform Kalman filter with model error, referred to as ETKF-Q, in the latent space.

The first part of this approach is to replace the expensive time integration of the model with a NN surrogate. A key question that is raised is how to ensure the time stability of the resulting scheme when the model is repeatedly called to propagate the state over several time steps. To keep this scheme stable an explicit stabilization is introduced into the training loss that applies a penalization of the growth of model iterations.

Another feature of this approach is that it performs data assimilation directly in the NN latent space, where it an underlying geometry and lead to the performance of computations mostly in the space where

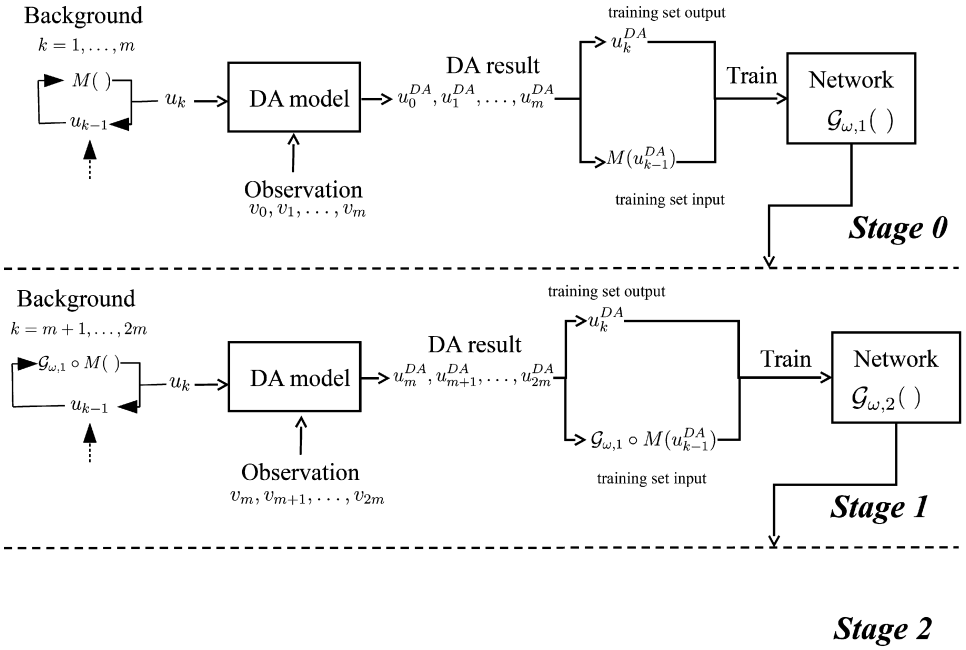


FIGURE 24.8

Copy of Figure 4 from [14] showing the schematic of the DDA scheme.

they are cheap. They do state that to obtain good results, special care has to be taken in the description of the dynamical and observational error. This DA system is referred to as ETKF-Q-L. Important points that are made about this approach are:

1. Explores the ability of DL to create a ℓ-dimensional reduced space, based on the assumption that a latent space of size ℓ that accurately represents the full dynamics exists. This is achieved with an autoencoder.
2. Defines a surrogate network within the latent space to perform the time propagation. An innovative iterative training approach enforces the surrogate to be stable over time.
3. Implements an ensemble DA algorithm within the learned ℓ-dimensional latent space thanks to the AE and the surrogate network.

As highlighted in [334], the first two of these can be performed in an all-at-once approach by training both the AE and surrogate at the same time using a well-suited custom loss function. The training set is an ensemble of simulations of the physical system that lies in $\mathbb{R}^N$. Thus, the proposed methodology provides the following advantages:

1. Since any DA algorithm requires the storage of vectors lying in the model space, discovering a lower-dimensional representation induces a reduction in memory needs and computational cost.

2. Performing the DA linear analysis in the latent space obtained by AE is less susceptible to yield nonphysical solutions since the decoder is a nonlinear transformation that fits the manifold where the state trajectory statistically belongs, when such a structure exists.

We have a copy of Figure 4 from [334] in Fig. 24.9 that details how latent DA works compared with the regular DA: since a latent dynamics exists in $\mathbb{R}^{\ell}$, latent DA leverages the nonlinear transformation provided by the encoder, whereas full space DA might not capture the intrinsic dynamics and yields a poorer estimate.

There are extensive description of the data assimilation algorithm in latent space, as well as a demonstration of this new approach with the Lorenz 96 model in [334].

24.4.4 Deep Learning for Fast Radiative Transfer

While not a direct data assimilation algorithm, radiative transfer models (RTM) plays a vital part in our ability to directly assimilate satellite radiances. In [412] they state that physics-based line-by-line radiative transfer models fulfill the requirement of accuracy of the RTM, but are too slow and costly in computational terms for operational applications. Therefore, fast methods were developed to be able to perform fast RT calculations, and the current operational configurations calculate the absorption and scattering coefficients from the pre-computed regression coefficients, atmospheric state and cloud profiles. In [412] they investigate a deep learning approach to replace the regression coefficients in the fast RTM, where a selection of hidden-layer NN configurations are trained against atmospheric transmittance profile data computed by an accurate line-by-line model.

In [412] they present two schematics of the NN, the first is for the input layers where we have temperature profile, pressure profile, humidity profile, and a CO_2 profile. We have a copy of figure 3 from [412] in Fig. 24.10.

One of the things that is investigated in [412] is the efficiency of either: 1) using a separate NN for each pressure level, or 2) using a global NN for the entire atmospheric profile for all layers simultaneously. To comprehend what is involved with respect to the NN for approach 1 as it is referred to as in [412], they provide a schematic of the NN in their figure 4, which we have a copy of in Fig. 24.11. The

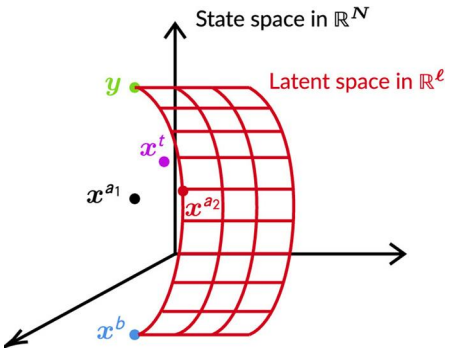


FIGURE 24.9

Copy of figure 4 from [334].

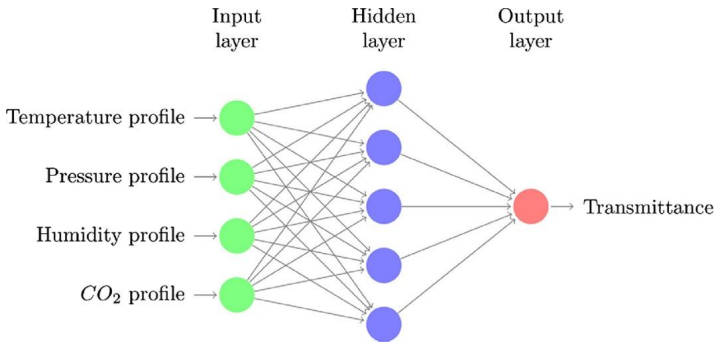


FIGURE 24.10

Copy of figure 3 from [412].

paper provides details of the performance of both approaches as well as listing their advantages and disadvantages, and the reader is referred to [412] for those details.

24.4.5 Using ML to Correct Model Error

In [118] they state a Bayesian framework for machine learning and DA. They start with the system state $\mathbf{x}_k \in \mathbb{R}N$ at discrete time $t_k \in \mathbb{N}$, and suppose that the evolution of the state is governed by

$$\mathbf{x}_{k+1} = \mathcal{M}_k^t(\mathbf{x}_k), \tag{24.26}$$

where $\mathcal{M}_k^t$ is the resolvent of the unknown true dynamical model from t_k to t_{k+1}. The objective is provide a surrogate to the true model, the resolvent of which can then be used to predict $\mathbf{x}_{k+1}$; given $\mathbf{x}_k$.

A standard ML approach to this problem consists of minimizing the cost function:

$$J(\mathbf{p}) = \frac{1}{2}\sum_{k=0}^{N_t-1} \|\mathbf{x}_{k+1} - \mathcal{M}_k(\mathbf{p}, \mathbf{x}_k)\|^2_{\mathbf{Q}_k^{-1}} + \mathcal{L}(\mathbf{p}), \tag{24.27}$$

where N_t us the length of the training trajectory, $\mathbf{x} \mapsto \mathcal{M}_k(\mathbf{p}, \mathbf{x})$ is the resolvent of the surrogate model from t_k to t_{k+1}, $\mathbf{p}$ is the set of coefficients used to define the surrogate model, i.e. weights and biases of a neural network, and $\mathcal{L}$ is a regularization term. In [118] they state that minimizing (24.27) amounts to finding the surrogate model that fits the trajectory best, and is equivalent to supervised learning.

The main drawback of this approach is that, for realistic applications, the true state of the system $\mathbf{x}_k$ is only known through the observation vectors $\mathbf{y}_k$. Thus (24.27) needs to be expanded to the more general cost function

$$(\mathbf{p}, \mathbf{x}_{0:N_t}) = \frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}_0^b\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\sum_{k=0}^{N_t} \|\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)\|^2_{\mathbf{R}_k^{-1}} + \frac{1}{2}\sum_{k=0}^{N_t-1} \|\mathbf{x}_{k+1} - \mathcal{M}(\mathbf{p}, \mathbf{x}_k)\|^2_{\mathbf{Q}_k^{-1}} + \mathcal{L}(\mathbf{p}), \tag{24.28}$$

where $\mathbf{x}_{0:N_t}$ is the system trajectory $\{\mathbf{x}_k, k = 0, 1, \ldots, N_t\}$.

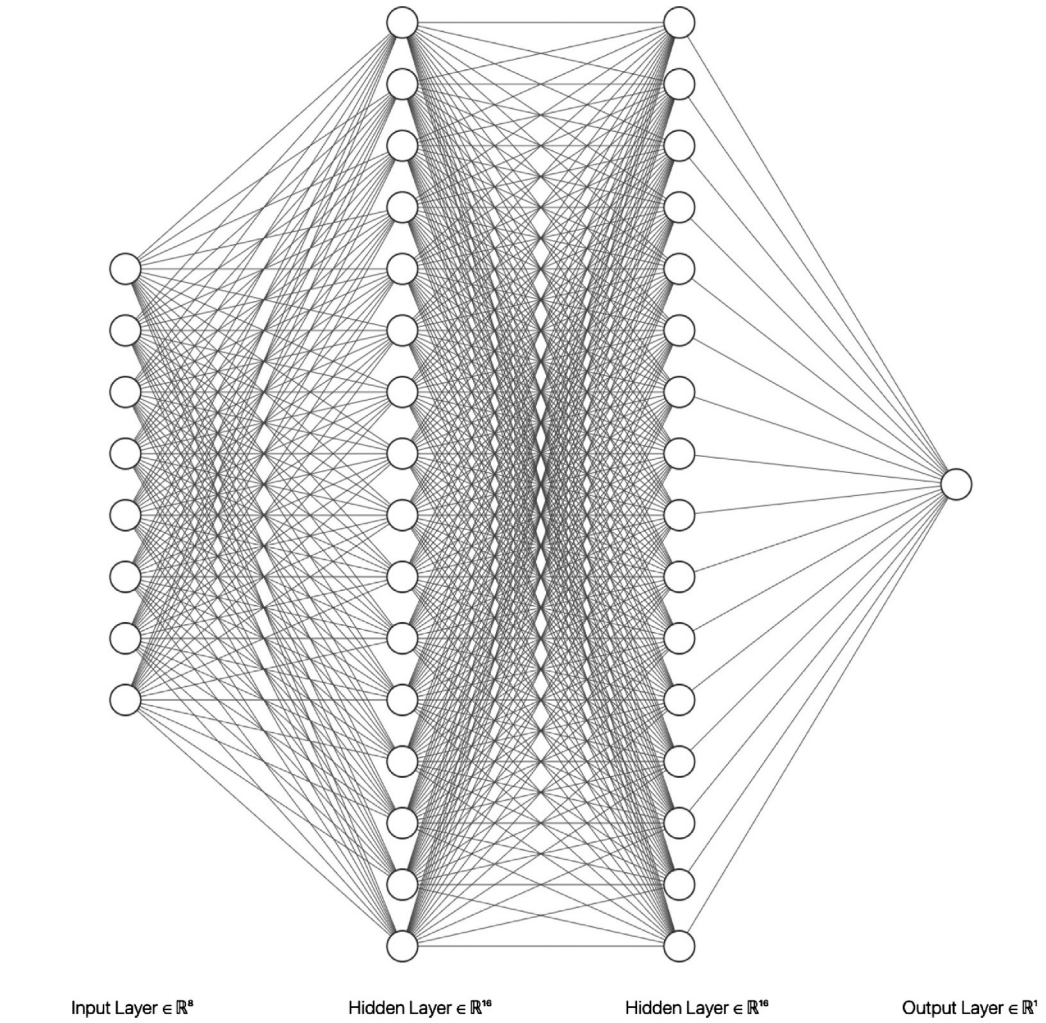


FIGURE 24.11

Copy of Figure 4 from [412].

Given this set up there are two possible surrogate models that are considered in [118]: correcting the resolvent, and correcting the tendencies. For the correction to the resolvent approach, the resolvent of the hybrid surrogate model from t_k to t_{k+1} can be written as

$$\mathcal{M}_k\left(\mathbf{p},\mathbf{x}\right) \equiv \mathcal{M}_k^o\left(\mathbf{x}\right) + \mathcal{M}_k^{ml}\left(\mathbf{p},\mathbf{x}\right), \tag{24.29}$$

where $\mathcal{M}_k^o$ is the resolvent of the original model and $\mathcal{M}_k^{ml}\left(\mathbf{p},\mathbf{x}\right)$ is the ML model. Note that the correction has been added to the original resolvent, which means that the output of the ML model lies in

state space. This choice has the advantage of being simple from a methodological point of view, while having the potential to handle any model error. However, other choices are possible, for example multiplying the original resolvent by the correction term or even composing the original resolvent and the ML model. The latter choices would be more appropriate if the true model error has a simpler definition in multiplicative form $\frac{\mathcal{M}^t}{\mathcal{M}^o}$ composition form $\mathcal{M}^t \circ (\mathcal{M}^o)^{-1}$ (provided that these quantities exist) than in additive form $\mathcal{M}^t - \mathcal{M}^o$. Thus substituting (24.29) in the cost function in (24.28) yields

$$\begin{aligned}\left(\mathbf{p}, \mathbf{x}_{0:N_t}\right) = &\frac{1}{2}\|\mathbf{x}_0 - \mathbf{x}_0^b\|^2_{\mathbf{B}^{-1}} + \frac{1}{2}\sum_{k=0}^{N_t}\|\mathbf{y}_k - \mathbf{h}_k\left(\mathbf{x}_k\right)\|^2_{\mathbf{R}_k^{-1}} \\ &+ \frac{1}{2}\sum_{k=0}^{N_t-1}\|\left(\mathbf{x}_{k+1} - \mathcal{M}_k^o\left(\mathbf{x}_k\right)\right) - \mathcal{M}_k^{ml}\left(\mathbf{p}, \mathbf{x}_k\right)\|^2_{\mathbf{Q}_k^{-1}} + \mathcal{L}\left(\mathbf{p}\right). \end{aligned} \tag{24.30}$$

Two points should be highlighted. First, the DA steps (except for the first one) are performed using the hybrid model defined by (24.29). If the ML model $\mathcal{M}_k^{ml}$ is implemented using standard ML tools, NNs, then there is almost no technical difference between the first DA step, performed using the original model only, and the following DA steps, performed with the hybrid model. In particular, if needed (for example, when using variational DA), the adjoint and tangent linear (TL) of the trainable model can be obtained through the ML library. Second, the ML steps are similar to a standard ML step, the difference being that the ML model has to learn the error of the original model, which has consequences in the preprocessing stage only.

In summary, [118] state that this method requires only minor modifications to the existing numerical methods, and should therefore be easy to implement.

If we now consider the correction being applied directly to the trajectory, then the surrogate model can be defined by the ordinary differential equation

$$\frac{d\mathbf{x}}{dt} = \mathcal{F}^o\left(\mathbf{x}\right) + \mathcal{F}^{ml}\left(\mathbf{p}, \mathbf{x}\right), \tag{24.31}$$

where $\mathcal{F}^o$ is the tendency of the original model and $\mathcal{F}^{ml}$ is the ML model. These hybrid tendencies must then be integrated from t_k to t_{k+1} to form the resolvent of the surrogate model $\mathcal{M}$.

This formulation essentially leaves (24.28) unchanged. However, the tendencies of the original model and the trainable model become intricate as a result of the integration and this has implications for the variational calculus. First, the contribution of the ML model to the adjoint and TL model of the resolvent of the surrogate model (potentially needed for variational DA) is nonlinear. Second, the gradient of the resolvent of the surrogate model with respect to $\mathbf{p}$ (usually needed for the ML steps) is not trivial, and in particular it may depend on the TL model of the resolvent of the original model.

In summary, and by contrast with the previous formulation, this method may require some substantial modifications to the existing numerical methods, but it has the potential to make predictions at the exact same horizon as the original model [118].

In [118] they test these approaches with the quasi-geostrophic model, or the Eady model when we introduced it earlier. The consider two different NN setups: a dense NN (DNN), and a combination of convolution layers and dense layers to form a CD-NN. One of the goals of this work is to predict where the model error is likely to be, and in [118] they produce a plot of the projected model error, figure 8 in [118], and we have a copy of this prediction in Fig. 24.12.

We note here that there is another manuscript by most of the same authors applying this technique in a more advance model in [117] and the reader is referred there for those details, as well as to [118] for the exact design of the experiment.

24.4.6 k-Nearest Neighbor for Data Driven Data Assimilation (DD-DA)

In [115] they state that in the EnKF, the forecast values are obtained from a physical model and so the general idea of the DD-DA approach is to replace the dynamical model applied to each ensemble member by a data-driven model instead. To show the difference between these two approaches Figure 1 in [115] is a schematic to illustrate this.

The data-driven model here comprises of a dataset. This data-driven model is composed of representative data catalog sets. For each continuous state variable, the catalog contains analogs and successors, that are formed by pairs of consecutive state vectors and separated by the same time interval. The second component of each pair is called the successor of its analogs. The catalog can be obtained from observations or numerical simulations, where the latter is used in [115].

The data driven aspect of machine learning to replace deterministic models is a growing field of research, and there is even talk of it replacing data assimilation, but that would not be for a while as the training required to understand the whole atmosphere is not possible as we do not have the data to start with, and as we know the chaotic behavior is very difficult to predict.

24.4.7 Other Applications

In this last subsection, we provide brief examples of the use of AI and DA.

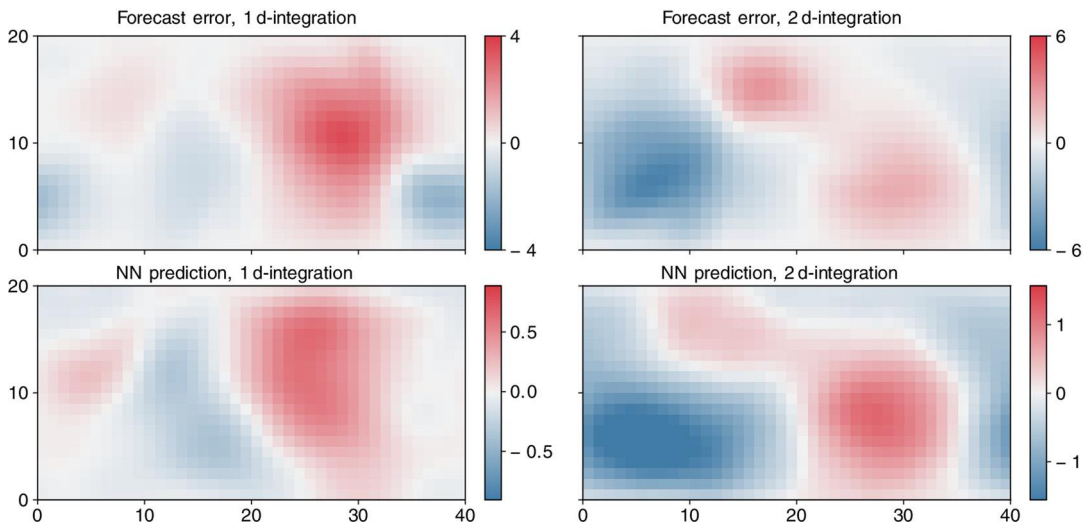


FIGURE 24.12

Copy of Figure 8 from [118].

Air pollution

In [355] they introduce a reduce order deep data assimilation model, referred to as RODDA. The DA model ingests information from observed data in the simulation provided by the Computational Fluid Dynamic (CFD) model. The results of the DA are used to train a NN learning a function which predicts the misfit between the results of the CFD model and the DA model. Thus, the trained function is combined with the original CFD model in order to generate forecasts with implicit DA given by neural network. Due to the time complexity of the numerical models used to implement DA and the neural network, and due to the scale of the forecasting area considered for forecasting problems in real case scenarios, the implementation of RODDA mandated the introduction of opportune reduced spaces.

In [355] they apply RODDA to a CFD simulation for air pollution, using the CFD software Fluidity, in South London, United Kingdom. They show that, using this framework, the data forecasted by the coupled model CFD+RODDA are closer to the observations with a gain in terms of execution time with respect to the classic prediction–correction cycle given by coupling CFD with a standard DA. Additionally, RODDA predicts future observations, if not available, since these are embedded in the data assimilated state in which the network is trained on.

We have a copy of figure 9 from [355] that shows the original mode output, the observations, the results from only using DA, and then from using RODDA in Fig. 24.13.

Ground water

In [255] they applied machine learning algorithms based on SVMs, combined with an ENKF to predict the change in groundwater levels (CGWLs) at 1 to 3-month time scales for 46 GW wells located at the northeast United States. The in-situ climate variables and the Gravity Recovery and Climate Experiment

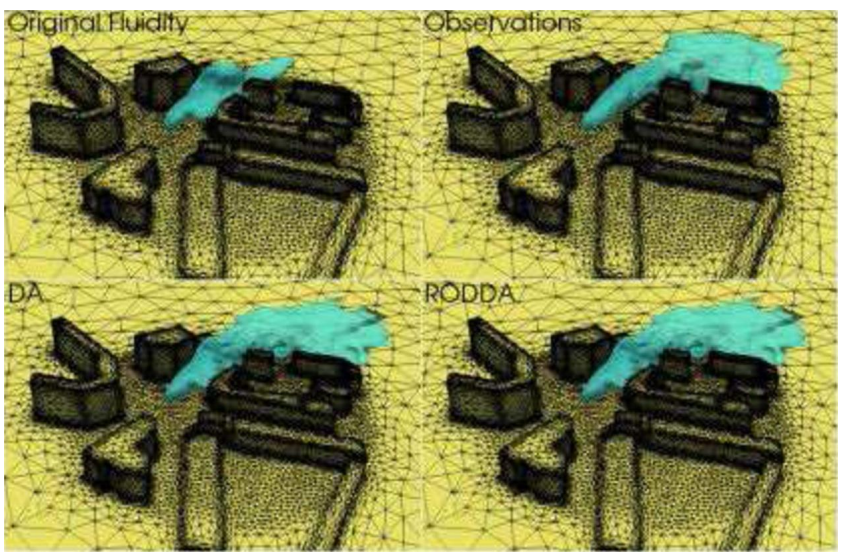


FIGURE 24.13

Copy of Figure 9 from [355].

(GRACE) mission-informed groundwater anomalies data (GWA) are used to develop the models. The results suggest that SVMs (SVM-DA) models forced with limited climate variables: precipitation, solar radiation, air temperature, infrared surface temperature, can forecast the CGWLs up to 3-month lead times at most of the locations. The addition of GRACE data as a forcing variable can improve the performance of SVMs at most of the stations, where a strong relationship exists between the CGWLs and the GWA. The SVM-DA model comparatively performed better than SVMs at most of the stations.

We have a copy of figure 3 from [255] that shows the flowchart of the SVM model as well as the SVM-DA model and how they interact with each other in Fig. 24.14.

Estimating of parameters of a convective-scale model

In [249] they look at the problem of parameter estimation through an artificial intelligence lens by training two types of artificial neural network (ANN) to estimate several parameters of the one-dimensional modified shallow-water model as a function of the observations or analysis of the atmospheric state. Through perfect model experiments they show that Bayesian neural networks (BNNs) and Bayesian approximations of point estimate neural networks are able to estimate model parameters and their relevant statistics.

The estimation of parameters combined with data assimilation for the state decreases the initial state errors even when assimilating sparse and noisy observations. The sensitivity to the number of ensemble members, observation coverage and NN size is shown. Additionally, they use the method of layer-wise relevance propagation to gain insight into how the ANNs are learning and discover that they naturally

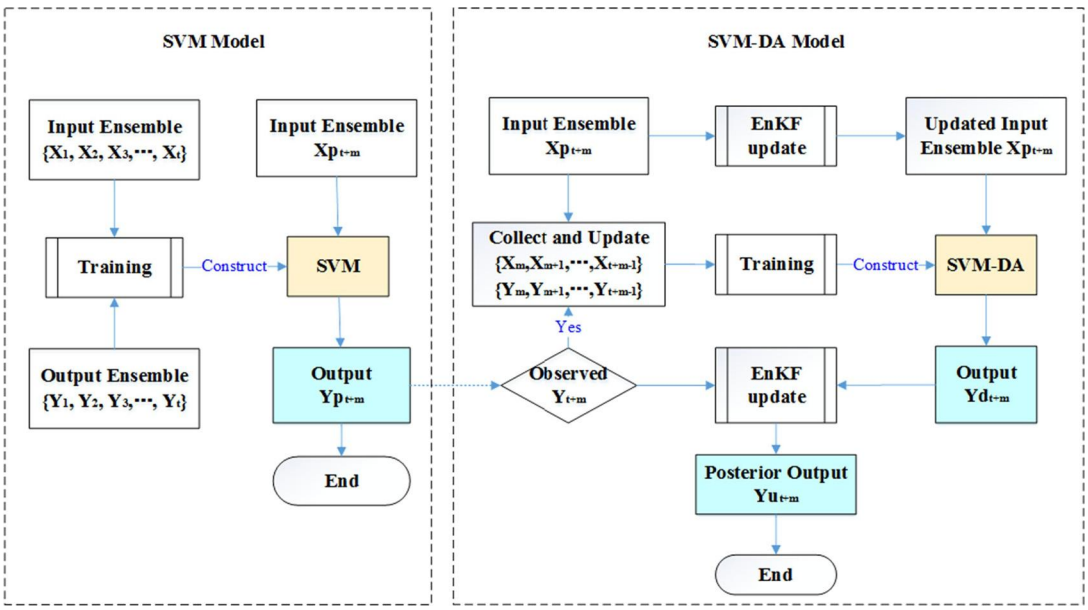


FIGURE 24.14

Copy of figure 3 from [255].

select only a few grid points that are subject to strong winds and rain to make their predictions of chosen parameters

NN-based observation operator

In [415] they are motivated to avoid the tangent linear and adjoint (TLM/AD) calculation involved in variational data assimilation. This intrinsically hampers the use of complicated observations. In [415] they assess a new data-driven approach to assimilate acoustic underwater propagation measurements, transmission loss (TL), into a regional ocean forecasting system. These measurements depend on the underlying sound speed fields and their inversion would require heavy coding of the TLM/AD of an acoustic underwater propagation model. In this study, the nonlinear version of the acoustic model is applied to an ensemble of perturbed oceanic conditions. TL outputs are used to formulate a NN operator. For the latter, two linearization strategies are compared, the best-performing one relying on reverse-mode automatic differentiation. The new observation operator is applied in data assimilation experiments over the Ligurian Sea (Mediterranean Sea), methodology to assess the impact of TL observations onto oceanic fields. The use of the NN observation operator is computationally affordable, and its general formulation appears promising for the adjoint-free assimilation of any remote sensing observing network.

As NN is a nonlinear model, it is not directly usable in variational data assimilation, whose cost function requires linear observation and model functions to preserve its quadraticity and, thus, the uniqueness of the analysis solution. Therefore, a tangent-linear approximation has to be formulated for the NN. A simple way to derive the tangent-linear of the neural network observation operator is through numerical differentiation, provided that the size of our problem (input and output variables) is much smaller than the data assimilation problem itself, and the prediction step of the neural network algorithm is relatively fast from a computational point of view. To this end, in [415] they tuned the Richardson's extrapolation algorithm implemented for numerical differentiation. It is an iterative method where the step size, i.e., the temperature increment in the evaluation of the transmission loss predicted by the neural network, is decreased sequentially and the derivative of the neural network is extrapolated for the increment tending toward zero. For more details about this part we refer the reader to [415], and to view figure 5 in [415] for the schematic of this approach.

Sub-seasonal forecasting

In [470] they present an ensemble prediction system using a Deep-Learning Weather Prediction (DLWP) model that recursively predicts six key atmospheric variables with six-hour time resolution. The model uses CNNs on a cubed sphere to produce global forecasts. Ensemble spread is primarily produced by randomizing the CNN training process to create a set of 32 DLWP models with slightly different learned weights.

24.5 Summary

In this chapter we have introduced some of the basics to understand the different terminology in papers and textbooks associated artificial intelligence. We have introduced different machine learning technique, both supervised and unsupervised as well as the deep learning neural networks. We have seen that AI is the overarching umbrella to describe algorithms that allow machines to develop patterns and

predictions, and recall that all deep learning is machine learning but not all machine learning is deep learning.

We have presented several examples of there the ML and DL techniques have been used in conjunction with data assimilation algorithms, or have been used to provide more accurate estimates of radiative transfer coefficients to improve the use of satellite radiance observations. DL techniques have been used to create observation operators that would have been too nonlinear to be used in a variational data assimilation technique, but as a result of this we are able to improve ocean data assimilation.

We make a passing reference here in the summary to some very intriguing work coming out of Imperial College, London, where they have created what they refer to as **Data Learning**, which is defined as integrating data assimilation and machine learning, [51]. This paper came out just as we are finishing this edition, but the reader is referred to this paper to see these really interesting ideas.

AI is here to stay and is a very powerful tool that can process data and build nonlinear relationships that could take a long time to derive analytical expressions, but even then these could be too nonlinear to be used in data assimilation algorithms of that they could slow them down to not be operationally viable.

This is the end of the theory chapters in this edition of the textbook. This edition of the textbook has two new theory chapters: Lagrangian DA, and AI-DA, along with the new advances in all aspects of the data assimilation theory throughout. We now move on to show the updated applications of data assimilation in the geosciences.

This page intentionally left blank

CHAPTER

Applications of Data Assimilation in the Geosciences

25

Contents

Data assimilation plays an indirect part in nearly everyone's lives, quite often without them even realizing, from the weather forecast that you see on your phone, computer screen, on television or read in

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00029-2

a newspaper, which could determine the plan for your day, to storm prediction for commercial flight paths, to whether a solar flare is going to affect telecommunications. Data assimilation can also indicate if a flash melting of snow could occur, and if so what the impact on the river systems would be, as well as whether or not this could cause flooding downstream.

In this chapter we shall summarize how data assimilation is used in a wide variety of geophysical situations. We shall show applications from the different scales of atmospheric data assimilation from the synoptic to cloud resolving and aerosol prediction, from global ocean data assimilation to the local bays of specific areas, as well as sea ice prediction. We shall present different forms of data assimilation in hydrology and cryosphere, as well as for space weather. We will introduce the Joint Effort for Data assimilation Integration (JEDI), along with Observing-System Experiments (OSEs), along with Observing System Simulation Experiments (OSSEs). We shall also present summaries of how data assimilation is being used in the optimization of the product of renewable energy. We review an area of increased interest in the data assimilation community which is the area of coupled data assimilation. This is where two or more different geophysical models are running, quite often different forms of data assimilation. The goal of coupled data assimilation, instead of using the output from one system as the boundary condition for the other system, is for the different systems to work together to produce a consistent analysis for all of the models involved.

We start with the discipline that has often led the way in the development of data assimilation theory, along with the ocean sciences, which is **atmospheric science**.

25.1 Atmospheric Science

We have seen that quite a lot of the development of data assimilation, not only in theory, but also in applications, has come from scientists working at operational numerical weather prediction centers. From the time of its formation in 1980, the European Center for Medium Range Weather Forecasting (ECMWF) has led the way, but the United Kingdom's Met. Office (UKMO) was not far behind, nor was the National Center for Environmental Prediction (NCEP), along with the Naval Research Laboratory (NRL) and Météo-France (MF), the Meteorological Service of Canada (MSC), NASA Goddard Space Flight Center's Global Modeling and Assimilation office and the Bureau of Meteorology Research Centre, and the Japanese Meteorological Agency.

Many different scales are involved in numerical weather prediction, but there are also other processes in the atmosphere associated with the synoptic scales. In this section we summarize the application of data assimilation at the synoptic, mesoscale, and cloud-resolving scales, along with aerosols prediction.

25.1.1 Operational Numerical Weather Prediction Centers

When considering the synoptic scale, we have to realize that there are different forecast length times that we wish for the forecast that has been initialized from the analysis. As the name suggests, EC (Medium-Range) WF is focused on producing forecasts that are as accurate as possible in the medium range. The medium range is considered to be 3–8 days, possibly up to 10. Naturally the short-range forecast needs to be as accurate as possible, otherwise this may reduce the accuracy of the medium-range forecast.

At the UKMO, their remit is for the short range, which is typically the 0–3 day range. Given the desired forecast range accuracy, different features need to be resolved. At the 3–8 day range you will focus more on resolving the movement of the pressure systems; while you will be producing smaller-scale features due to the resolution of the model, the major driving forces are the larger scales in the atmosphere. In the short range, on the other hand, you could be focused on frontal passages, as well as individual convective storms, but also rapidly developing cells that could cause major infrastructure damage, as well as the possible loss of human and animal life. Many of the operational numerical weather prediction centers run forecasts up to about 6 days and are focused on the shorter ranges of prediction, but most centers cover a full range of scales up to the climate scale.

ECMWF

At ECMWF they currently run their synoptic scale numerical model, which is a spectral-based approximation in the horizontal, with an approximate resolution of 9 km, and 137 vertical levels. Pressure at the model top is 0.01 hPa, and the vertical component is a finite element model. The model just described is referred to as the **Integrated Forecast System (IFS)**. This system produces forecasts for up to 10 days. Its data assimilation is a hybrid incremental 4D VAR system, but the hybrid component is an ensemble of data assimilation systems. The system runs 50 incremental 4D VARs in parallel at an 18 km horizontal resolution. The ensemble is initiated through perturbing the observations that all of the 4D VARs assimilate. We should also note that ECMWF runs with three outer loops with plans to jump to as many as five.

The assimilation window length for the 4D VAR system is 12 hours; this is due to ECMWF using weak constraint 4D VAR, that attempts to compensate for the model error which then enable the tangent linear model assumption to hold for such a long time. ECMWF runs two 4D VAR analysis steps a day; the two assimilation runs are separated by 12 hours of initialization time, but are both minimizing over 24-hour windows.

United Kingdom's Met. Office

As we mentioned earlier, the Met Office is tasked with producing short-range forecasts, but it uses a multiple time scales in its numerical weather prediction suite. Here we focus on the short-range forecast, which is the 0–3 day range. The current Met. Office's numerical model is a grid point approach known as a **unified model (UM)**, and has a horizontal resolution at the mid-latitudes of 10 km. The UM has 2560 by 1920 grid points and 70 vertical levels. Assuming the same number of grid points on each vertical level puts the total number of points in the grid at 344,064,000 up from 123,863,040 grid points in the first edition. We must recall that we have more than one variable at each of these points and as such there are nearly two and a half billion entries in the state vector.

The Met. Office runs an incremental hybrid 4D VAR, where they employ the α control variable for the coupling of the error covariance approximations from the ensemble members, where the approximation come from the analysis of an ensemble transform Kalman filter (ETKF) scheme. The ensemble system which is in the En4D VAR section now contains a control member with 17 perturbed members to 7 days, there is a 36 member ensemble generated by time-lagging over 12 hours, effectively the last two analysis cycles, at a 20 k resolution, where there are 1280 by 960 grid points in the horizontal and with 70 vertical levels; as noted above, these increments come from the ETKF.

Météo-France (MF)

The MF data model configuration is quite similar to that of ECMWF, in that they use a spectral model for the horizontal and finite elements in the vertical. However, their grid configuration is quite different. The grid that is used in their data assimilation system applies a Schmidt projection, which moves the poles to a different location. This enables a higher resolution in the horizontal to be obtained over a different region that would be in the poles. As a result of the pole being over Paris, MF is able to have a horizontal resolution of approximately 7.5 km over France and most of Europe, while having a resolution of approximately 35 km in the South Pacific. They run with a 105 vertical levels. A hybrid of 4D VARs is used to form their own hybrid 4D VAR system, similar to ECMWF.

NOAA-NCEP

Since the first edition of this book NCEP has undergone a major change to its numerical weather prediction system. It now uses what is referred to as the FV3, which is the finite volume, 3 dimensions, developed at NASA, on a cubed sphere. NCEP implemented an operational version of 4DEnVAR in May 2016 with their spectral numerical weather prediction model and have successfully transition this to the FV3 dynamical core on the cubed sphere. This model is also referred to at the Unified Forecasting System, UFS, and is available to the general public.

MSC

The synoptic scale data assimilation system that is used at MSC, which is part of Environment Canada, is also a 4DEnVAR system. In fact the Canadians were the first to implement the four-dimensional version of EnVAR in Fall 2014 [49]. An interesting feature about the EnVAR implementation in [49] is that their hybrid system is a two-way coupling where the ensemble Kalman filter (EnKF) assimilates observations with quality control and bias correction performed by the Global Deterministic Prediction System (GDPS) (the 4D VAR component). The GDPS then passes to the ensemble of background states produced by the EnKF to specify the background error covariance for the 4DEnVAR. The ensemble component of the hybrid system is known as the Global Ensemble Perdition System. Fig. 25.1 illustrates how the coupled hybrid system works.

NRL

The operational global data assimilation system developed at NRL is quite unique in that it uses the dual formulation. This means that the data assimilation system is solving for the optimal increment in observation space. The cost function that is minimized is that of the Inc 4D VAR, but through the method of accelerator representers. The numerical model is a spectral-based model, but where the analysis increments from the data assimilation system are found on the 1 degree Gaussian grid. The data assimilation scheme is also a hybrid scheme where the information from an ensemble is used to approximate the analysis covariance matrix [226]. The Navy is in the process of transitioning to a new model called NEPTUNE and looking to use a configuration of the JEDI system in model space. JEDI will be introduced in the next section.

25.1.2 Limited Area Synoptic Scale Data Assimilation

While in the previous section we presented a summary of some of the world's operational numerical weather prediction center's synoptic model, albeit the resolution of the global scale models are now at

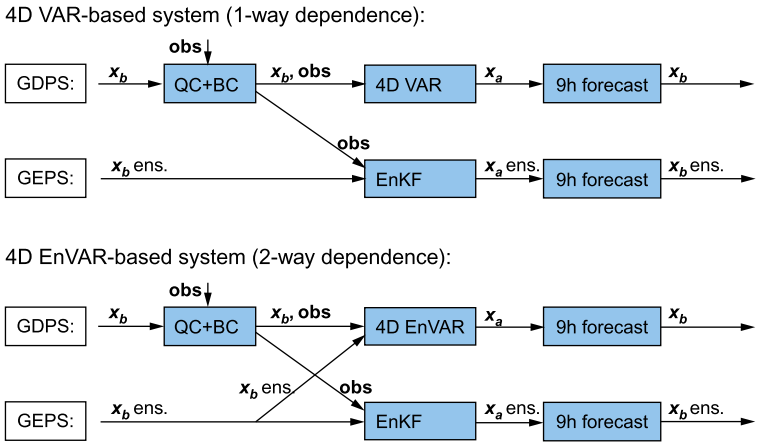


FIGURE 25.1

Schematic of the MSC hybrid data assimilation system, which is a copy of figure 1 from Buehner, M., McTaggart-Cowan, R., Beaulne, A., Charette, C., Garand, L., Heilliette, S., Lapalme, E., Laroche, S., Macpherson, S. R., Morneau, J., and Zadra, A. (2015). Implementation of Deterministic Weather Forecasting Systems Based on Ensemble–Variational Data Assimilation at Environment Canada. Part I: The Global System, Monthly Weather Review, 143(7), 2532-2559. **© American Meteorological Society. Used with permission.**

the mesoscale, they are still the global model. There are data assimilation schemes that are used with limited area models, but are resolving synoptic-scale features, and we consider a case with tropical cyclone prediction below.

Tropical cyclone prediction

One such situation is for the limited area modeling of hurricanes. A hurricane-based data assimilation system requires data to be assimilated such that the tropical cyclone is of the right intensity and size; however, it is also important for it to be in the correct place, and for the forecasted track to be accurate.

It could be the case that a tropical cyclone could match two of the three static components, e.g., correct size and intensity at time t, but in the wrong place. This will cause problems for a data assimilation system, as it will imply that we have a non-Gaussian observational error as we will have a bimodal distribution, or the root mean square error (RMSE) will be seen as punishing you twice, but really you have two errors: one at the location of the storm is in the model and one where the observations/true feature is.

Nehrkorn et al. developed a technique known as *correcting for position error* [318]. Errors caused by a feature being incorrectly positioned can be described in terms of displacements. Therefore the techniques introduced in [318] solve for the displacements, where is it assumed that the residual errors after the aligning of the background field will be smaller, but that they could be more Gaussian in terms of distribution as well.

While this technique has been introduced into the Weather, Research, and Forecasting (WRF) data assimilation system, we voice some caution here; by displacing a feature and smoothing the environmental fields surrounding the feature's new and old locations, we have introduced a drastic change into

the numerical model. As such, does the system need to do extra work to keep the model in balance to stop spurious gravity modes? Another concern about this technique is that it does not address the bigger question: why does the model have the feature so far away? This indicates a model error that should be compensated for, or that there is a larger problem in the numerical model that needs to be corrected. While we know that data assimilation is only supposed to introduce small increments, if we have such a large model error, we have to address the cause of that error.

The technique introduced in [317] works by introducing the displacements, $\mathbf{s}$, as control variables. In [318] the authors define $\mathbf{s}$ as the concatenation of the horizontal displacements at all the grid points, and as such the associated change in the model state $\mathbf{x}$ is defined as

$$\delta \mathbf{x}_s^n = S\left(\mathbf{s}; \mathbf{x^{n-1}}\right), \tag{25.1}$$

where S is the model alignment function that depends on the displacements and the constant model guess state $\mathbf{x}^{n-1}$. The aligned values of the model state at horizontal location $\mathbf{r}$ are obtained by horizontally interpolating the model state variables to the source point of the displacement vector as

$$S\left(\mathbf{s}; \mathbf{x^{n-1}}\right) = \mathbf{x}^{n-1}\left(\mathbf{r} - \mathbf{s}\right) - \mathbf{x}^{n-1}\left(\mathbf{r}\right). \tag{25.2}$$

The authors then introduce a linearized version of the nonlinear alignment function [317], denoted by $\mathbf{S}$ such that the observational component of the cost function becomes

$$J_{os} = \frac{1}{2}\left(\mathbf{d} - \mathbf{H}\left(\mathbf{U}\mathbf{v^n} + \mathbf{S}\mathbf{s}\right)\right)^T \mathbf{R}^{-1}\left(\mathbf{d} - \mathbf{H}\left(\mathbf{U}\mathbf{v^n} + \mathbf{S}\mathbf{s}\right)\right), \tag{25.3}$$

where the term $\mathbf{Uv}$ is the control variable with its transform. To solve for the displacements [318] introduces a third term into the 3D VAR cost function as we did for the α control variable.

From a mathematical point of view this is a good technique, but we are dealing with a highly nonlinear geophysical system, and as such there could be feedbacks that we are not aware of. Also, if we are applying this technique to a tropical cyclone, then the feature is dependent on the coupling with the sea surface temperature, so there will be a feedback there. If we are not coupled to the ocean model but we move a tropical cyclone, then the Sea Surface Temperatures (SST) underneath may not be consistent with the atmospheric feature above or be able to support it.

25.1.3 Mesoscale Data Assimilation

Most, if not all, of the operational centers we mentioned in the last subsection run a form of mesoscale data assimilation systems. These systems are usually part of a subdomain of the global data assimilation system and are usually run over their specific country or over a specific region, either local or an area of interest. Throughout the second half of the book, we have made reference to the WRF model and one of its data assimilation systems, the Gridpoint Statistical Interpolation (GSI). WRF-GSI is a capable hybrid 3D VAR First Guess at Appropriate Time (FGAT) system that can be located anywhere in the world.

Most mesoscale data assimilation systems use a form of nested approach, where the *parent* numerical model is ran at a lower resolution than the resolution of the model where the assimilation took place. This is to ensure that lateral boundary errors do not affect the assimilation scheme in the area of

interest. As these mesoscale models are **limited area models**, they require lateral boundary conditions from a larger-scale model.

An example of using the ETKF-3D VAR WRF hybrid can be found in [392], where the authors assimilate radar radial velocity data with respect to the prediction of Hurricane Ike in 2008. They illustrate the impact that the static, isotropic, homogeneous correlations have on the increments to the 500 hPa temperature fields from using only the 3D VAR system compared to the hybrid system, where flow dependencies come into the analysis and you can clearly see the difference between the two fields. They also show results for the 850 hPa temperature increments, where there is a dramatic difference between the two approaches.

In [499], the authors apply a 3D VAR assimilation scheme over a region of southern China, where they assimilate wind profiling radar observations to assess the impact on convection permitting quantitative precipitation forecasts. They show that through assimilating these radar data, they are able not only to reduce predicted spurious precipitation events, but also to produce better 6-hour heavy rain forecasts. Therefore, the application of data assimilation here has two positive effects: (1) it reduces the number of false alarms, which then builds confidence and trust by the general public that the over-forecast of precipitation has been reduced; and (2) data assimilation enabled the model to provide more accurate short-range forecasts of heavy precipitation, which then enables the public to be aware of possible severe weather.

In [35] the authors use the Local Ensemble Transform Kalman Filter to assimilate 3D radar reflectivities into the COnsortium for Small scale MOdelling Convection, permitting the numerical weather prediction model of Germany. They indicate that the assimilation of the radar reflectivities improved short-term prediction. They showed by assimilating the 3D radar reflectivities that they were able to improve the precipitation location and significantly improved the forecast out to 4 hours, which again created reliability in the forecast.

25.1.4 Cloud Resolving Data Assimilation

There is presently a lot of research underway into the possibility of assimilating cloud infected/cloudy radiances into all scales of numerical weather prediction. The technique of being able to use cloudy radiances is referred to as **all sky** assimilation. However, we then have to include some form of representation for the cloud variables into the control vector. We also need a reasonable representation of the cloud-scattering properties in the radiative transfer model. A full cloud resolving full field 4D VAR was developed at the Cooperative Institute for Research in the Atmosphere that was based on the Regional Atmospheric Modeling Systems or RAMS, which is also the mascot of Colorado State University where RAMS was developed. The fill field 4D VAR system is referred to as RAMDAS. It contains a full moist adjoint of the model for all seven classes of hydrometeors that RAMS predicted in 2005, and these are: cloud water, rain, graupel, hail, aggregates, pristine ice, and snow. RAMDAS employs a detailed form of Hessian preconditioning [511].

RAMDAS has been used to assimilate cloud-affected infrared radiances for the GOES-8 satellite in [341], where the authors investigated the effect of the maximum allowed residual, the length of the window, and the inclusion of other ground-based observations, as well as introducing a cloud mask, which appears to have had a positive impact.

However, as mentioned earlier, it was shown using RAMDAS in [391] that if the background state does not have an initial cloud with which to start the minimization with respect to, then there will be

no sensitivity in the adjoint of the radiative transfer model to the hydrometeors. In addition, as we mentioned in the least square section, inverse problems have a tendency to be non-unique and that is what we saw in the work in [391], where we could create the same brightness temperature through adjusting the environmental variables, albeit resulting in an unphysical solution.

More recently there has been work to assimilate AMSR2 microwave all sky radiances, focusing on the forecast of Hurricane Sandy with a limited area WRF-GSI [493]. The authors demonstrated that the assimilation of the cloudy radiance was able to produce a significant reduction in the track error as well as a better prediction of the central sea level pressure of the storm. It was also shown in [493] that the authors could produce a better 72-hour forecast as a result of the assimilation of the all sky AMSR2 radiances. Below we have a copy of figure 5 (Fig. 25.2) from [493], where we can clearly see the significant difference between the clear sky and the all sky brightness temperature equivalent.

It is not just the application of the variational and the ensemble Kalman filters that can provide us with information and give us guidance on model development. The Markov chain Monte Carlo (MCMC) method can also give us information, rather than a forecast that is vitally important in developing consistent models to use with forecasts. In [342], Posselt uses the MCMC method to examine the sensitivity of a deep convective squall-line to changes in cloud microphysical parameters.

In [491] the authors apply a 3D VAR system for the assimilation of Weather Surveillance Radar-1988 Doppler (WDR-88D) with an extremely high-resolution model referred to as the Advanced Regional Prediction System where they are running with four nested grids at 9 km, 1 km, 100 m, and 50 m grid spacing. The reason for this work was to see the impact of the radar observations in the prediction of the mesocyclone that formed and led to the May 8, 2003 Oklahoma City tornadic supercell. In [491] the authors were able to show that through the use of data assimilation, they were able to generate a tornado on the 50 m grid quite similar to the one that occurred in 2003. This is an important application of data assimilation, as in the Great Plains areas of the United States it could lead to better advance warning of a tornado before they form.

25.1.5 Retrievals

While retrievals do not usually produce a forecast, they do provide vital information about the geophysical variables that are of interest to us. The National Oceanic and Atmospheric Administration, or NOAA, has an operational retrieval system called the Microwave Integrated Retrieval System or MiRS, which produces images of the globe through assimilating the radiances from multiple microwave sensors. MiRS has been an all sky retrieval system since 2011 [46]; in [46] it is stated that they invert the radiances to obtain estimates of: cloud liquid water, ice water path, land surface emissivities, land surface temperature, liquid water path, moisture profiles, rain rate, rain water path, snow cover, sea ice concentration, surface type classification, snow water equivalent, temperature profiles, and total precipitable water. As we can see this is an extensive list of geophysical variables that could be of use to multiple different research activities.

25.1.6 Atmospheric Chemistry and Aerosols Assimilation

In this section, we consider some applications of data assimilation to a problem that is probably highly non-Gaussian. The first application of data assimilation that we consider here is the estimation of volcanic ash emissions. During April 2010, the Eyjafjallajökull volcano erupted on Iceland, sending huge volumes of volcanic ash up into one of the world's busiest flight zones, affecting Heathrow, Gatwick,

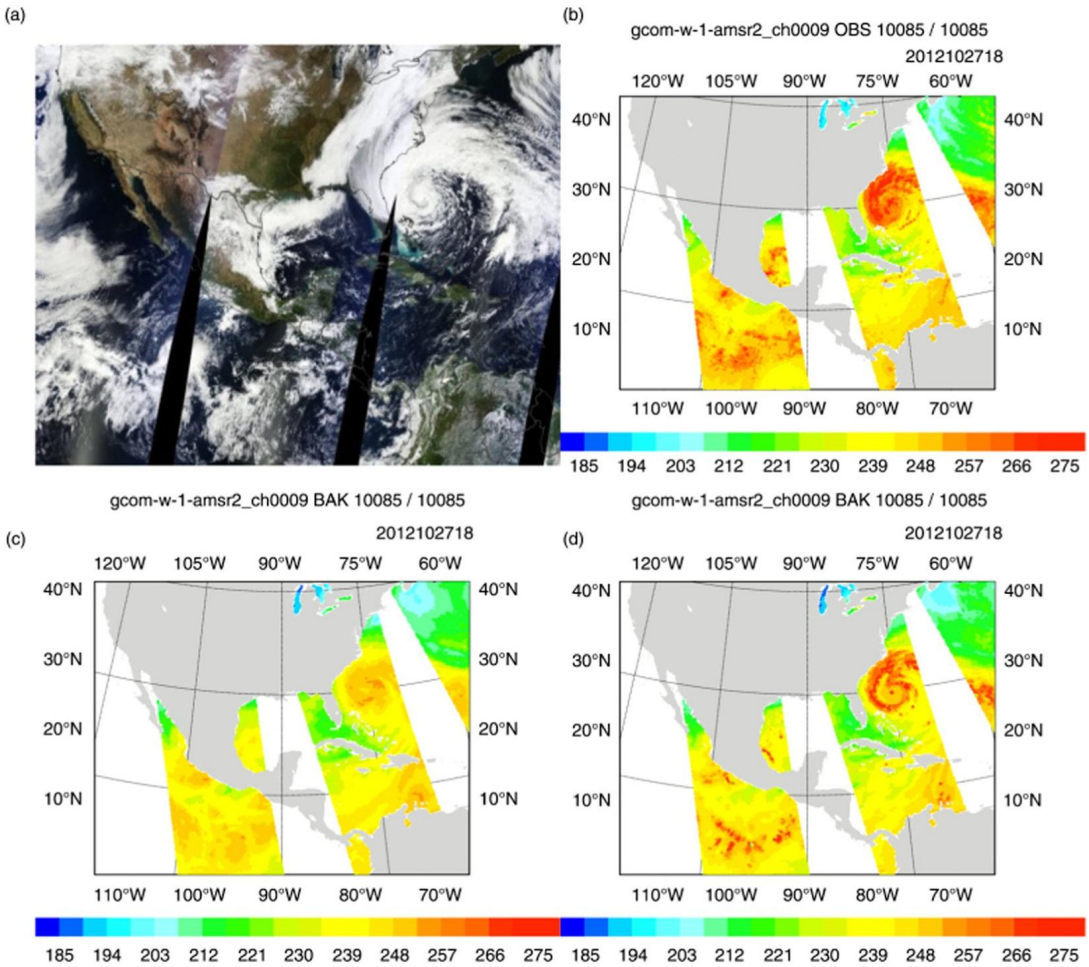


FIGURE 25.2

Copy of figure 5 from [493] which is of the MODIS image of Sandy and then the equivalent brightness temperatures from the AMSR2 sensor, and then from the different data assimilation experiments. Chun Yang, Zhiquan Liu, Jamie Bresch, Syed R. H. Rizvi, Xiang-Yu Huang and Jinzhong Min (2016) AMSR2 all-sky radiance assimilation and its impact on the analysis and forecast of Hurricane Sandy with a limited-area data assimilation system, Tellus A: Dynamic Meteorology and Oceanography, 68:1, DOI: 10.3402/tellusa.v68.30917. https://creativecommons.org/licenses/by/4.0/.

Paris, and Frankfurt, to name but a few airports. The volcano sits on the flight path between Europe and North America. The Northern Atlantic airspace was closed for 5 days due to the concern that the floating ash, which contains silicate, could be ingested by commercial and military aircraft engines and cause devastating risk to the jets.

One of the problems associated with this event was that there was no real way of knowing where the ash was. In [272] the authors introduce what they refer to as trajectory 4D VAR, where they reformulate the cost function in a regression type that computes the total difference between observed ash columns and a linear combination of simulated trajectories coupled with a priori emission knowledge. The cost function is constructed through decomposing the system into subsystems that represent source-receptor relationships to each source term (ash injected at each layer).

In [272] the authors show that they are able to predict more accurately the movement of the ash cloud from the April eruption compared to 4D VAR; see Fig. 25.3.

As mentioned above, the ability to predict volcanic ash is of huge benefit to society, but [272] showed that is it possible to adapt the variational-based schemes due to the fact that they are seeking the most likely state given the prior and observational information.

More recently there has been work to assimilate volcanic aerosols using the FALL3D dispersal model and the Parallel Data Assimilation Framework (PDAF), where the DA system here is the LETKF assimilating satellite-retrieved column mass loading, [302] but that this approach leads to nonphysical solutions, indicating that a non-Gaussian based approach could be better suited.

One of the important aspects of aerosol data assimilation is in the operational need for mass transport modeling and forecasting relative to viability forecasting which plays a vital role, again in aviation. In [497] the authors present results assimilating a suite of satellite observations into the US Navy's Navy Aerosol Analysis and Prediction System (NAAPS), which used NAVDAS to form NAVDAS-AOD, where AOD stands for Aerosol Optical Depth. Note that it is a physical space analysis system-based system. The authors are able to show that by assimilating the complete suite of satellite data that they consider, they are able to reduce the observation minus analysis errors quite substantially, which leads to the analysis being more trustworthy. Fig. 25.4 reproduces figure 5 from [497], and we can see the drastic improvement in the O-A scores, and the difference in the 48-hour forecast O-F for the AOD variable.

The local ensemble transform Kalman filter (LETKF) has also been used to better improve the forecast of AOD. In [495] the authors use the LETKF with the Spectral Radiation-Transport Model for Aerosol Species. They showed that through using the data assimilation method, they were able to predict dust transport more accurately over North Africa as well as the Middle East region.

25.2 Joint Effort for Data Assimilation Integration (JEDI)

As we have seen data assimilation is an essential component of any forecasting system. It aims at determining the best estimate of the state of a system given observations of the system and a previous estimate of the state of the system for use as initial condition for an ensuing forecast. It also requires error estimates of the input quantities, usually covariances matrices and bias estimates, and uses operators, such as the forecast model and observation operators.

The Joint Effort for Data assimilation Integration (JEDI) framework is being developed at the Joint Center for Satellite Data Assimilation (JCSDA), where its data assimilation system is centered around the Object Oriented Prediction System (OOPS) that defines abstract representations of the quantities and operators used in data assimilation, with the various operations that can be performed with them. This abstract interface layer is implemented in C++ using templates. Different implementations are chosen at compile time during template instantiation and provide genericity across models. Adding a

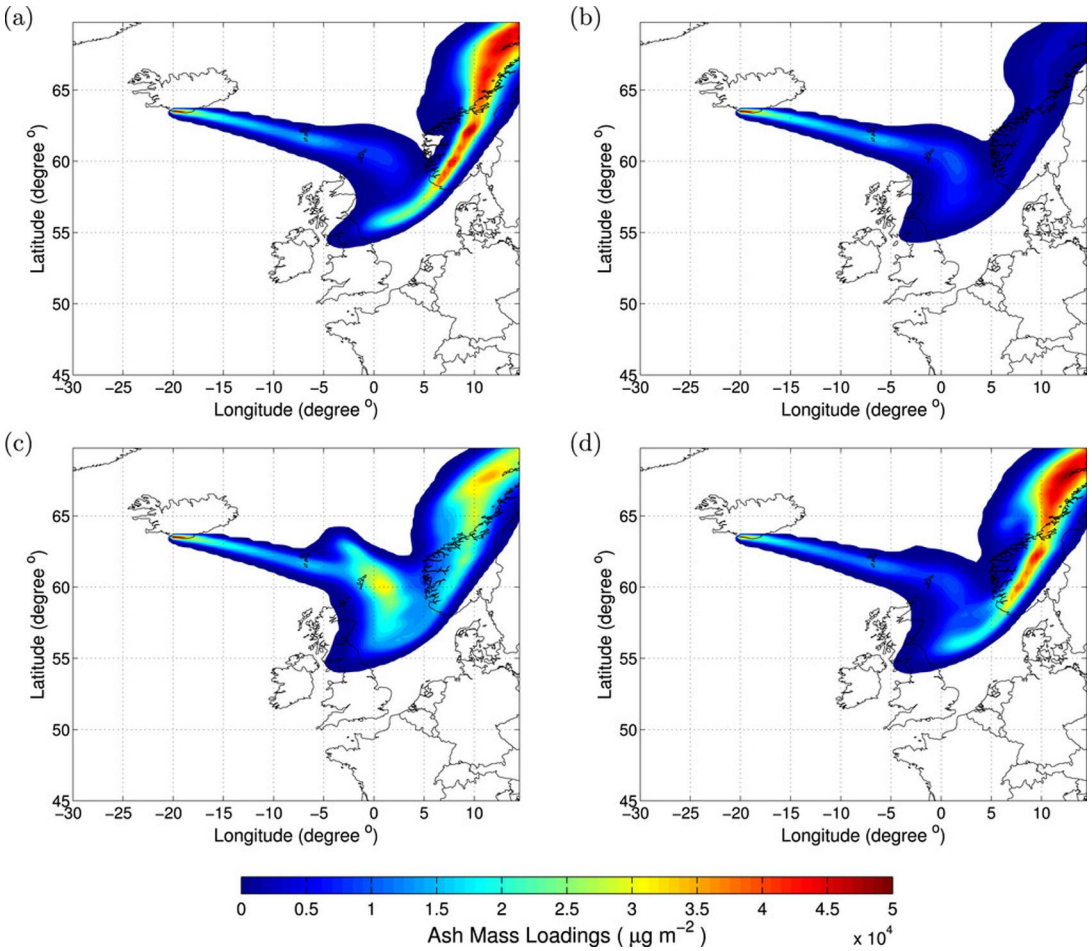


FIGURE 25.3

Copy of figure 10 from [272] which is of the different forecasts of the ash cloud after the different forms of data assimilation have been applied. Lu, S., Lin, H. X., Heemink, A. W., Fu, G., and Segers, A. J. (2016). Estimation of Volcanic Ash Emissions Using Trajectory-Based 4D-Var Data Assimilation, Monthly Weather Review, 144(2), 575-589. **© American Meteorological Society. Used with permission.**

new model or system to JEDI means implementing the classes defined in the abstract layer for that model or system. On the other side, adding or modifying a data assimilation algorithm is done through the interface layer so that all code at high level is model independent. What follows is a summary of the description of the JEDI system from the JCSDA's 66th Quarterly Newsletter, Winter 2020, available at https://doi.org/10.25923/rb19-0q26.

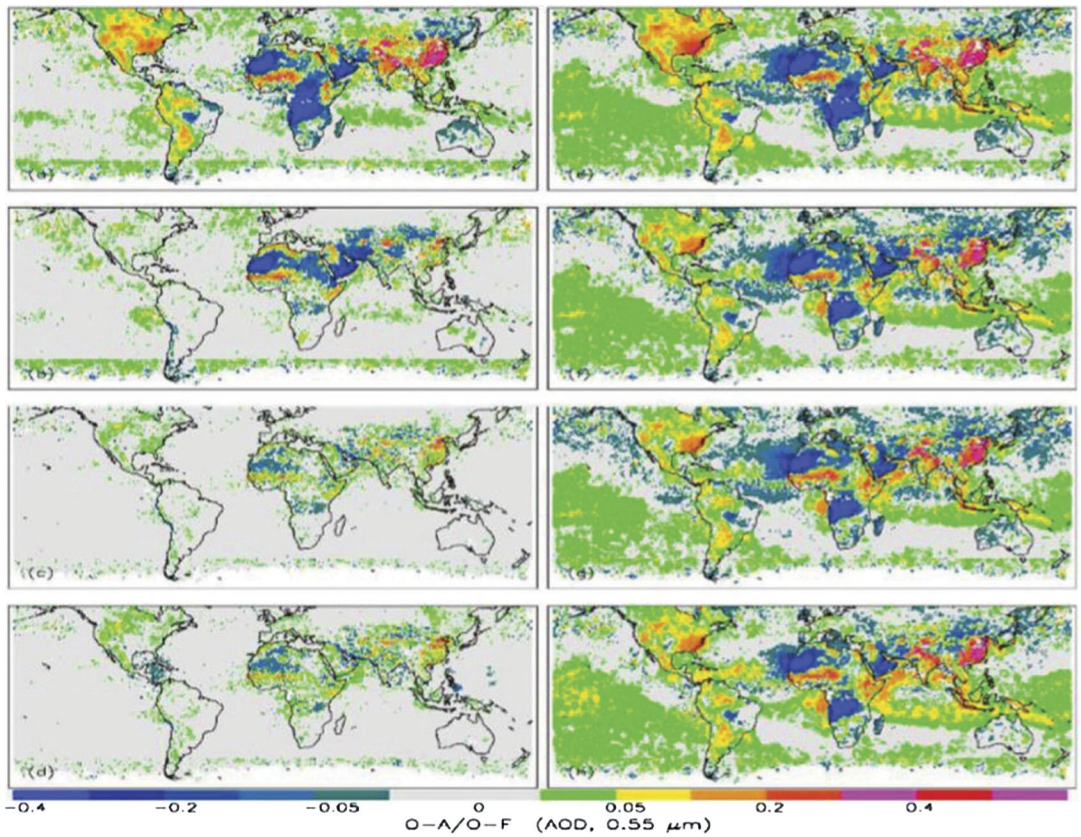


FIGURE 25.4

Copy of figure 5 from [497] which is of the mean relative differences between observed values and the analysis of the many different combinations of observations, along with the 48-hour forecast differences.

25.2.1 OOPS Abstract Interfaces

The starting point is the model space interfaces, where DA algorithms are described in terms of a state variable, $\mathbf{x}$, without any reference to the specific nature of the fields in $\mathbf{x}$ or the geometry of their discrete representation. In JEDI all such details are encapsulated inside a state class and are not visible from high-level algorithms. In a similar way, forecast models, $\mathcal{M}(\mathbf{x})$ are used to evolve the state of the system of interest forward in time, but the details of how this is done are not required in order to define a data assimilation or other high-level algorithms, Therefore in JEDI such details are not apparent in high level parts of the code.

Geometry

The state of the system is an important piece of data in any data assimilation and forecasting system, often represented by a collection of fields that are the values of the variables of the problem. For

computational purposes, fields are discretized and a finite set of values are stored and manipulated. This set of values is distributed on a **model-dependent geometry** for example on regular grids, cubed-sphere or reduced Gaussian grids, as well as possible as spectral representations. In OOPS the geometry class is dedicated to holding the definition of the model grid, resolution, and distribution across processors. OOPS creates Geometry objects and passes them to lower level code where necessary. Passing the same Geometry to the constructors ensures consistency in resolution and distribution across processors between the objects involved.

State and increment

The State class is the fundamental class giving access to operations on model states in OOPS. It holds and encapsulates data that define the state of the system and the date and time for which it is valid.

The Increment class is similar to the state class except for the fact that it handles perturbations to the state. It provides a method to compute an increment as the difference between two states, to add an increment to a state, and a set of basic linear algebra operators, that are legitimate operations for Increments but not for States.

Model

The Model class holds the forecast model configuration data and provides the ability to evolve a state in time. Its main method is the forecast method, that is coded in the OOPS layer and relies on specific model implementations to provide three lower level methods: **initialize, step, and finalize**. For a given model, these three methods contain respectively everything that happens before the loop over time steps, inside the loop, and after the loop.

For improved efficiency in the case several calls to the forecast are made in the same executable, code that should be executed only once should be in the constructor of the Model, while code that needs executing at the beginning of each integration should be in the initialize method.

The reason for this design of the abstract interface for the model is that 4D assimilation methods require access to the model state throughout the assimilation window. The volume of data that would represent far exceeds the amount of memory available on supercomputers, so it is not possible to store it in memory. For the same reason, I/O would be prohibitively expensive. These four-dimensional computations have to happen while the model is running. Some algorithms also require running the forecast and computations needed for data assimilation repeatedly in an iterative process. The level at which the interface between the model and the data assimilation is written in OOPS is the highest level possible that meets those requirements while remaining generic.

25.2.2 Observations Space Interfaces

Again as we know data assimilation algorithms are described in terms of a vector of observations, $\mathbf{y}$, without any reference to the specific nature of the observations in $\mathbf{y}$ or their distribution in space and time. These details are encapsulated inside an observations class and are not visible to high-level algorithms. In a similar way, observation operators, $\mathbf{h}(\mathbf{x})$, are used to simulate the observations given the state of the system $\mathbf{x}$, but the details of how this is done are not required in order to define a data assimilation or other high-level algorithms; thus, such details should not be apparent in high-level parts of the code.

Observations and departures

The Observations class holds and encapsulates observations and associated operations, whereas the Departures class is the mirror in observation space of the Increment class. It represents differences between, or perturbations to, observations. It is very similar to the Observations class except that it provides an additional set of linear algebra operators.

For data assimilation applications, there is one object of class Observations per observation type. There is no strong constraint in the system about what constitutes an observation type other than the fact that the same observation operator, quality control procedures, and bias correction method will be applied to all observations in a given type.

ObsSpace and ObsVector

The concept of geometry from the model space is transposed to the observation space with the ObsSpace class. This class defines the distribution of observations in space, as the Geometry would for model space entities. However, the ObsSpace class also gives access to all metadata associated with the observations, including time, some instrument dependent metadata, quality control information and to the observation values.

The ObsVector class is used to hold values in observation space. As State and Increment are always defined with a Geometry, ObsVectors always refer to an ObsSpace.

ObsOperator

The computation of a simulated observation given a model state comprises two steps: the interpolation of the model variables to the observation locations, and the computation of the observation equivalent from these interpolated model variables. It is the responsibility of the state to provide the values of its variables (or fields) at the requested locations through the getValues method, which isolates the observation part of the code from the internal geometry of the model being used.

The ObsOperator class define the computation of observation equivalents given model state values at the observations' locations for a given observation type. This step can be extremely simple if the quantity being measured is a variable of the model, or very complex in the case of radiance observations from satellites involving a radiative transfer model. This class encapsulates the science related to the observation type and isolates it from the technical details related to the forecast model.

The LinearObsOperator class holds the tangent linear and adjoint of the observation operator.

25.2.3 Error Covariances

As we know there are three sets of errors associated with data assimilation: background (forecast), observational, and model. There is also analysis error but that is associated post the solving of the DA problem. However, in JEDI only background and observational errors are tackled at the moment.

Background errors

The design of the interfaces for background error covariance matrices is quite simple. In addition to a constructor, the ErrorCovariance class requires a method to multiply an increment by the covariance matrix and another one to multiply an increment by the inverse of the covariance matrix. In real cases, this inverse is often very ill conditioned, the variational algorithms in OOPS only use it for diagnostics. Some applications require a method to generate a random increment according to the statistics distri-

bution described by the covariance matrix. OOPS implements a factory that lets uses choose a given covariance model at run time.

Observation errors

As we know the definition of the observational error is key to the quality of the analysis. The type of observation error covariance is configurable at run time, but in the initial implementation this is a diagonal observation error covariance matrix

25.2.4 UFO, IODA, and SABER

These three tools are considered part of the generic layer and are defined as follows

Unified Forward Operator (UFO)

UFO is at the heart of the JCSDA's missing and is the other major component of JEDI after OOPS. It implements generic observation operators. The key elements that make the observation operators generic are the classes that make the connection between the model space and the observation space; Locations and GeoVaLs.

Thus, UFO includes implementations of these two classes in addition to a collection of observation operators that users can leverage. The same UFO operators can be used in conjunction with several models regardless of the models' internal geometries.

In addition to the observation operators themselves, UFO also implements generic tools for quality control and variational bias correction, which are extremely important for operational centers.

Interface for Observation Data Access (IODA)

IODA is developed with the UFO to handle observation data. It provides functionality for I/O of observation data and in memory access. In the interface layer and traits, it implements the ObsSpace and ObsVector classes. IODA will facilitate the implementation of UFO. It will also facilitate the exchange of observations between centers for experimental studies, comparisons, or potentially for reanalyses. Data assimilation diagnostics are being developed based on IODA.

System Agnostic Background Error Representation (SABER)

SABER implements generic background error covariance matrices, among which is BUMP (Background error on Unstructured Mesh Package). After observation processing, the modeling of background errors is the most time consuming task in data assimilation.

There are several applications of JEDI since the newsletter that we have taken certain parts from here, and we recommend the reader to the newsletters as well as to the JEDI website https://www.jcsda.org/jcsda-project-jedi to learn more about this joint effort.

25.3 Observing-System Experiments (OSE)

Observing-system experiments, quite often abbreviated, or referred, to as OSEs, or one s, are a common tool for assessing the impact of a sensor or whole observation types in a numerical prediction system. These experiments are often framed as a data denial to determine implicitly the value of observations

already actively assimilated to see of quantify the degradation of skill wrought by their absence. Such OSE in the past have considered the impact of denying all satellite data into an operational numerical weather prediction system, [291]. Another OSE may involve denying a single sensor into the data assimilation system. A third option is the reverse where we have data addition, which can be applied when considering active assimilation of a new senor or data type.

In [99] where they are performing an OSE to determine the impact of microwave sounders have on the ECMWF near operational 4D VAR system. They indicate that the study they undertake was during a golden age for microwave sounding observations as there were eight different satellite with temperature sounding channels and eleven satellites with humidity-sounding channels actively assimilated in the ECMWF system.

However as stated in [99] a possible criticism, of OSEs is that it quite often the case that the background errors in each experiment are assumed fixed, despite the fact that they are a function of the observing system itself, that changes in an OSE. In [99] they assess the impact of adding more microwave sounders by the change in the height root mean squared errors at 500 hPa, given as a percentage difference relative to what they refer to as the no sounder control. We have a copy of Figure 3 from [99] in Fig. 25.5. There are also useful results from adapting the **B** matrix in the OSE presented in [99] that will be of interest to the reader.

In [114] is a very good discussion about observation impact metrics in numerical weather prediction. In [114] they indicate that data denial experiment (DDE) and the forecast sensitivity based observation impact (FSOI) measure two different aspects in the OSEs. DDE measures the impacts on forecast accuracy of removing observation types from the system; whereas FSOI measures the amount by which an observation type reduces the short range forecast error, within a system containing all the observation types.

In [114] a metric to measure the DDE that are evaluated by comparing the full forecast error for the full observing system with the forecast error for the system in which one observation type has been removed. The first metric introduced is the mean percentage error metric where the difference in error

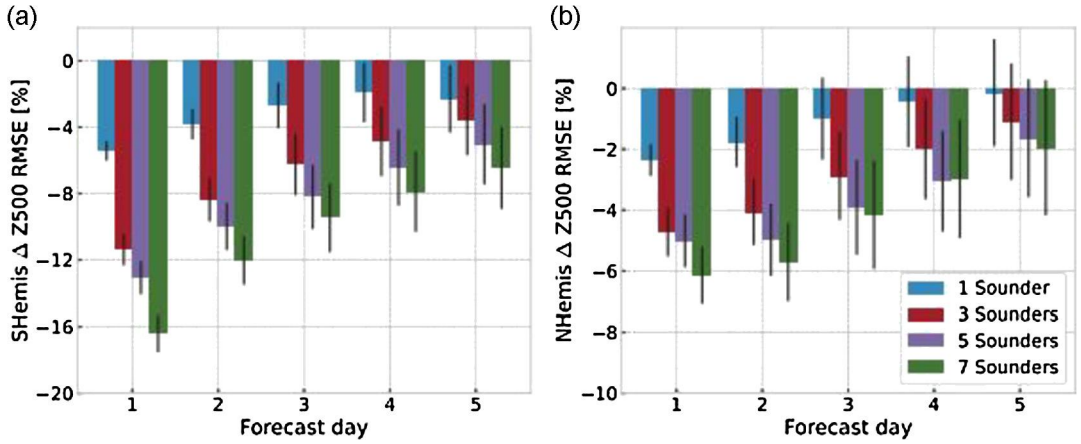


FIGURE 25.5

Copy of Figure 3 from [99].

variance between two runs for each variable is computed as a percentage if its values for the full system. The quantity is then averaged over all of the elements of **x**. The metric involves an approximation to the forecast error covariance matrix, simply denoted **P** here such that

$$\%DDE_n = 100 * \frac{1}{N} \sum_{i=1}^{N} \frac{\mathbf{P}_{ii_n}(deg) - \mathbf{P}_{ii_n}(full)}{\mathbf{P}_{ii_n}(full)}, \tag{25.4}$$

where deg is referring to the degraded system, while $full$ is referring to the full observing system. This does imply that two versions of the data assimilation system have been ran.

The other metric used in [114] is the DDE energy metric given by

$$\%DDE_n^e = 100 \frac{trace\left\{\mathbf{P}_{ii_n}(deg)\right\} - trace\left\{\mathbf{P}_{ii_n}(full)\right\}}{trace\left\{\mathbf{P}_{ii_n}(full)\right\}}. \tag{25.5}$$

In [114] they present the mathematics to quantify the difference between FSOI and DDE, and make the comment that it is quite often the case that in global NWP experiments, that the FSOI impact of a given observation type is greater than DDE, and the aim of [114] is to investigate this. To illustrate this point we have a copy of Figure 1 from [114] from a study experiment from the Met Office in the Fall of 2019 in Fig. 25.6. We will make the remark here that when some observations are removed from the assimilation, extra information could be extracted from other parts of the observing system.

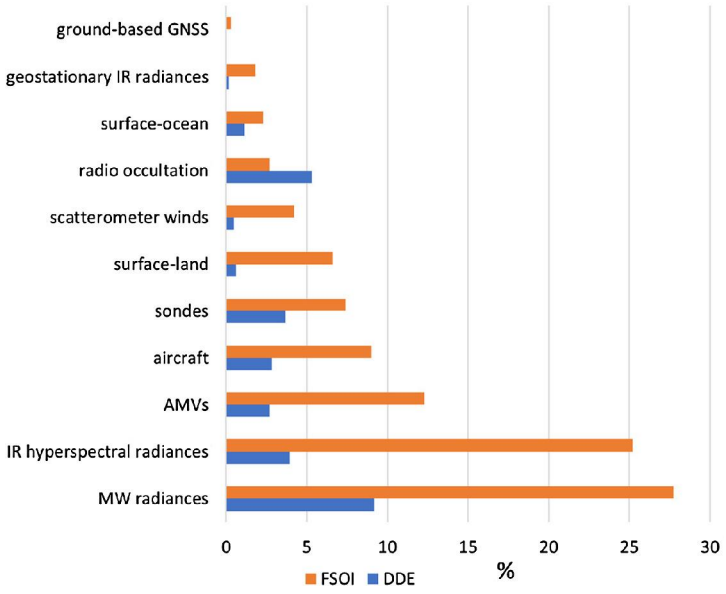


FIGURE 25.6

Copy of Figure 1 from [114].

As we saw earlier there is an equivalent to the FSOI for the ensemble based systems (ESOI), and we recommend the reader to [492] to see an example of an ensemble system in an OSE framework. There is also a very good example of the impacts being detected in an OSE for an ocean case in [215].

OSEs methods are a direct way of assessing impacts of observations on numerical prediction systems, note we have not tied anything here to the atmosphere or its numerical models. However, it can only be used with observations that are available from existing observing systems; it cannot be used to estimate the impact of future observing systems, This is address through using observing system simulation experiments (OSSEs), or two s, which we introduce in the next section.

25.4 Observing System Simulation Experiments (OSSE)

Observing System Simulation Experiments or OSSE as they are more commonly abbreviated to, can be very powerful tool to give insight of **possible** impacts of new observing system or observation type.

We start with the explanation of OSEEs from [108] which describes the initial design and **validation** of an OSSE at NASA's Global Modeling and Assimilation Office (GMAO). The opening paragraph of this paper states that an OSSE is a numerical experiment conducted with a data assimilation system (DAS) and numerical prediction model that traditionally uses simulated rather than real observations. These are drawn from some dataset representing the states to be observed. For an OSSE applied to the atmosphere, this is most appropriately a temporal sequence of atmospheric fields generated by a sufficiently realistic simulation model, termed a **nature run**. The simulated observations are then ingested by the DAS. Various metrics are applied to quantify the accuracy of the analyses produced, particularly standard ones measuring **fits to observations** and **forecast skill**. The impacts of various configurations of the observing system can then be compared. Unlike an OSE conducted with real observations, an OSSE is not limited to using observations that currently exist.

OSSEs are most often employed to estimate quantitatively the potential improvements in climate analysis and weather prediction to be gained by augmenting the present atmospheric observing system with additional envisioned types of observations that do not yet exist [108]. In particular, the utilities of competing designs of proposed observing systems can be compared within the context of modern data-assimilation systems. OSSEs can also be used to explore some otherwise elusive characteristics of an already existing observing system such as, for example, its analysis error statistics as in [107]. Essentially, these latter goals are achievable because in the OSSE framework a dataset representing the atmosphere or ocean is precisely known, unlike the case of the real atmosphere, thereby allowing an explicit and precise determination of analysis errors.

[108] now provides a warning to ensure that the integrity of the OSSE is maintained:

> Before conducting an OSSE to investigate proposed observing systems, there are several reasons to conduct baseline experiments simulating a current observing system. Most importantly, as a simulation, any particular OSSE framework should first be validated by comparing corresponding metrics of the DAS and subsequent forecasts applied to equivalent real observations in order to establish its credibility. Most proposed observing systems are intended to be augmentations of the existing (baseline) systems that they are supposed to improve, so a baseline validation is especially relevant. Many past OSSEs have been criticized privately because their validations have been insufficient or even absent, sometimes resulting in their gross misinterpretation.

As we mentioned at the beginning, an OSSE requires what is referred to as the nature run. The NR represents the *truth* for the OSSE and there are couple of different ones available for the atmosphere. ECMWF has one that is available, as does NASA, who most recent version is a 2 year period of record, run with NASA's Global Earth Observing System version 5 (GEOS-5)m that is a free run of the model at 7 km horizontal resolution with 72 vertical levels, and output saved every 30 minutes, [349].

Simulation of observations

In [108] they provide the following five caveats about how to generate the observations from the NR, and how to avoid extra representative errors:

1. The fidelity of some realistically simulated observations may be detrimentally affected by unrealism in some aspect of the NR (e.g. as will occur if radiances are computed using a good model for scattering by clouds but the high-level NR clouds are deficient);
2. The most realistic observation operators available may still be too physically deficient (e.g. although a radiance-scattering model may generally describe cloud effects well, it may treat poorly optically thin clouds that affect those radiance observations actually retained by quality-control (QC) algorithms);
3. Information on a data type or spatial scale required by a realistic observation operator may be absent from the NR (e.g. the cloud microphysical properties required by a radiation-scattering model);
4. The DAS may not consider effectively some aspects of the observations that characterize their realism, and thus great effort may be expended in simulating complex aspects for which a much simpler treatment may be adequate (as in the examples provided in what follows);
5. Some aspects of realism are expected to have little impact on the time- and space-averaged metrics to be provided either for this validation or for other studies planned for the near-term, and thus have little importance at this stage of development.

In [351] they explained that synthetic observations were generated based on the temporal and spatial distribution of real archive observations during the period of the NR being used. Conventional data types were created by interpolating the NR fields to the time and location of recorded observations. Radiance observations were created using the CRTM to generate brightness temperatures calculated from the NR fields.

The next step is to add the errors, and you wish for these to be as near to the real errors from the data assimilation systems as possible to ensure the impact being observed is not distorted by incorrect errors on the observations. In [351] they state that errors were added to the synthetic data to simulate a combination of observation error and representativeness error. Uncorrelated errors were added to all observations types, and an additional correlated component was added to some types. Vertically correlated errors were added to conventional sounding data types, horizontally correlated errors were added to AMSU, HIRS, and MSU observations, where channel correlation was added to AIRS, where vertically and horizontally correlated errors were added to satellite wind observations. No correlation was applied between observation types.

The importance of the simulated errors in OSSEs was explained very well in a recent, at time of writing, paper from Dr. Privé, Dr. Errico, and Dr. McCarty from NASA and can be found in [350]. If looking for more details about OSSEs these three names are a good place to start to look at their papers.

Validation

As stated in [352] validation plays an important part in an OSSE framework, considering that all aspects of the OSSE are simulated, but the results are used of that simulation to infer what occurs in reality.

In [349] they state that the metrics used include observation impacts relative to other data types, forecast anomaly correlations, and the root mean square forecast error, here validated against the NR. In [352] they include what is referred to as self-analysis, sometimes abbreviated to self-anal, pronounced self-a-nal. This is when a 24 hour or what ever length forecast, is validated against the analysis from that data assimilation system valid at the forecast end time.

OSSEs can be used for a multitude of applications and investigation. One of the advantages of the NASA NR is that there is an adjoint of the model, as such the FSOI can be applied, which is used in [352] to explore the impacts of observational data on forecasts in the 6-48 hour range, they perform validation against self-analysis and the true nature run. It is stated in the abstract of [352] that the verification against self analysis was found to inflate the estimated forecast error growth resulting in an overestimation of observation impacts. We have a copy of Figure 1 from [352] in Fig. 25.7 that indicates this effect.

We make two comments here: it is possible to perform and OSSE without an adjoint as is done in [103] where they use an ensemble transform sensitivity metric to assess the impact on winter storms forecasts it a satellite gap was replace with idealized targeted dropsondes. Second, OSSEs are not restricted to the atmosphere, in [163] they are using a nature run for the ocean in the Gulf of Mexico, and the North Atlantic to quantitatively assess rapid-response pre-storm ocean survey to improve coupled tropical cyclone prediction. These two nature runs, at the time of writing are available at https://cimas.rsmas.miami.edu/research/projects/ocean-modeling-osse-center/index.html.

A *nature run* has recently been developed for snow mission in [482] for land surface model based OSSE.

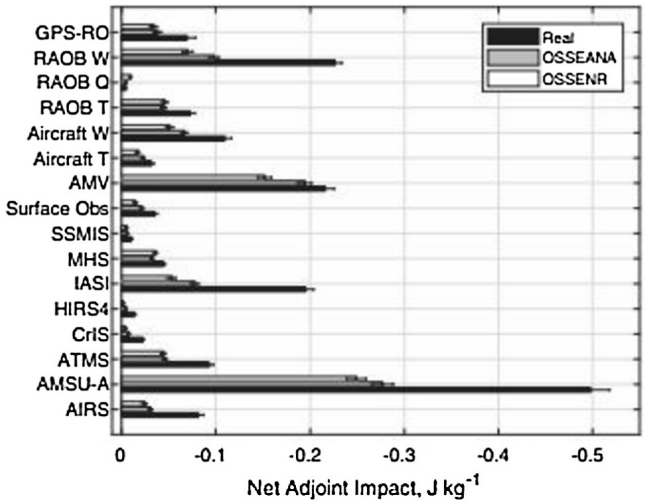


FIGURE 25.7

Copy of Figure 1 from [352].

25.5 Oceans

Most of the world's operational numerical weather prediction centers also have an operational ocean data assimilation system. These could be global for all the Earth's oceans or they could be for a specific region, or even a local bay.

25.5.1 Global Ocean Data Assimilation

In this section, we present an example of global ocean assimilation schemes with the Met. Office's ocean data assimilation system. The Met. Office has a variational-based data assimilation system that runs with the Forecasting Ocean Assimilation Model (FOAM). The model produces analyses and forecasts of ocean current, temperature, salinity, sea-surface height, and sea ice concentration, and these are used by Her Majesty's Royal Navy, commercially, for research, and for initializing the Met. Office's seasonal prediction system. The FOAM system uses the hydrodynamic model Nucleus for European Modelling of the Ocean (NEMO) and the new 3D VAR-FGAT that was implemented at the Met. Office in [467] is referred to as NEMOVAR. It assimilates satellite and in situ sea-surface temperature observations, in situ temperature and salinity profiles, altimeter sea-level anomaly observations, and satellite sea ice concentration. The Met. Office use observations from animal-borne sensors [58], where they assimilated temperature and salinity measurement from mammals (mostly elephant seals) to depths of up to 2000 m in the high latitude regions, where there are very few in situ observations.

Since the first edition there has been work on the NEMOVAR system to assimilate satellite-derived sea-ice thickness data from CryoSat-2 and SMOS in the Arctic, [297], where they show that assimilating CryoSat-2 observations the sea ice-thickness increments are generally positive in areas of thinner ice; however those increments become negative when the SMOS data is also assimilated. This is shown in Figure 3 from [297] and we have a copy of this figure in Fig. 25.8.

25.5.2 Regional Ocean Data Assimilation

In [315] the authors state that due to recent increases in the price of crude oil, efficient fishery operations are more important than ever. A complication that can affect the fisherman is as a result of strong currents at the desired depth that could prevent them from setting their fishing nets there. In [315] the authors state that they are able to run their nowcasting/forecasting system at 1-hour time steps with 1 km scales. The data assimilation system used is called the Data assimilation Research of the East Asian Marine System or DREAMS. The assimilation schemes is a reduced rank Kalman filter, which is used in the three subcomponents: DREAMS-B is a basin-scale model; DREAMS-M is a marginal ocean model; and DREAMS-I is the local coastal scale is at a resolution of 1.5 km.

Data assimilation took place in the DREAMS-B and DREAMS-M models, and then these feed the boundary conditions for the higher-resolution DREAMS-I system. Fig. 25.9 shows a copy of the description of the setup of the three domains from [315].

An interesting feature of the assimilation system in [315] is that the authors were able to obtain flow field data from observations of acoustic Doppler current profilers installed on commercial fishing boats. These new sets of observations help to determine that there was a submesoscale at approximately 10 km in the eastern boundary current region of the Japan/East Sea.

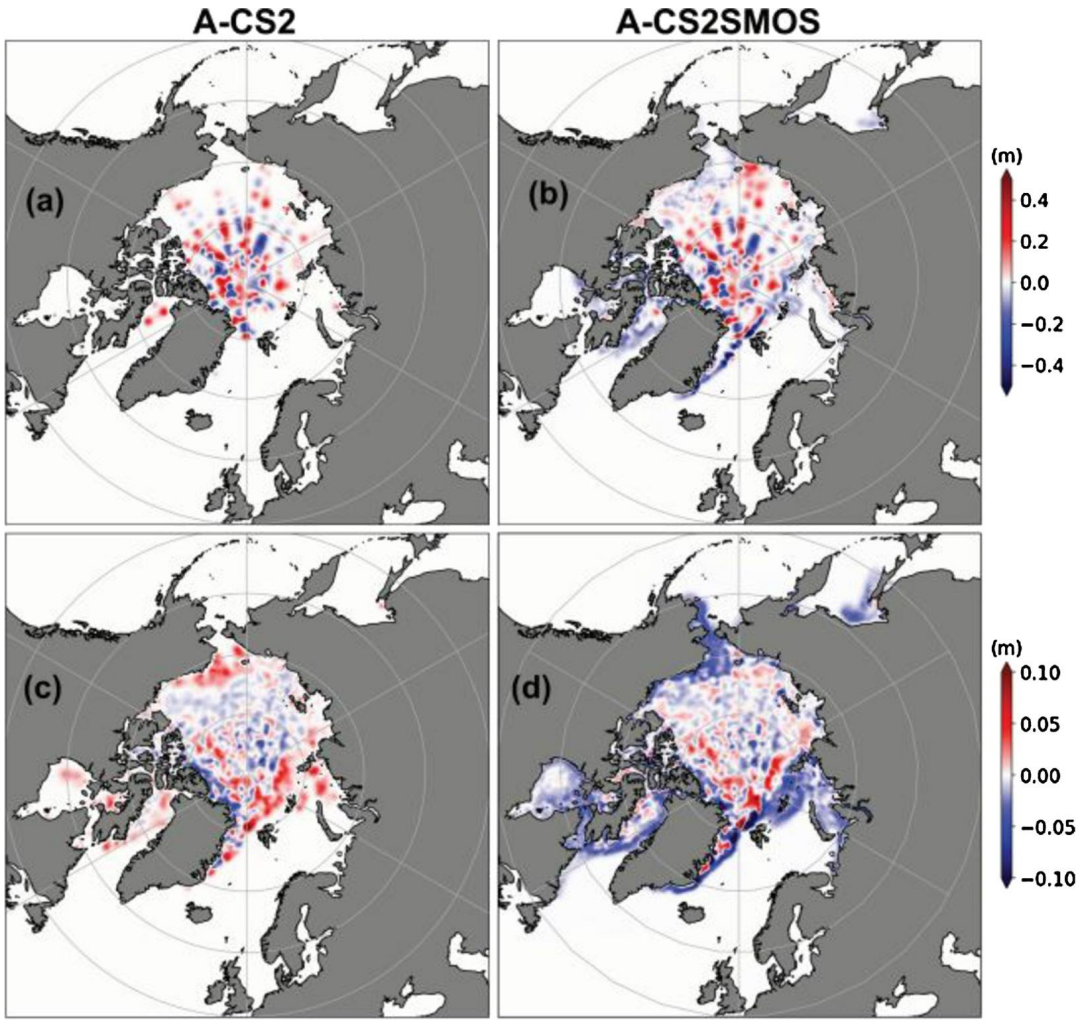


FIGURE 25.8

Copy of Figure 3 from [297].

This is an example of the application of data assimilation to help minimize the usage of fuel for fishermen through assimilating data into the model that then enabled fishermen to identify whether the current at the desired net depth was too strong.

An important component of the regional ocean data assimilation scheme is the ability to predict not only the current along the coast, but also the sea surface temperatures; if the ocean and atmosphere model are not coupled, then these act as boundary conditions for the atmospheric model. Another important feature that the ocean data assimilation schemes have to predict is the Gulf Stream, which

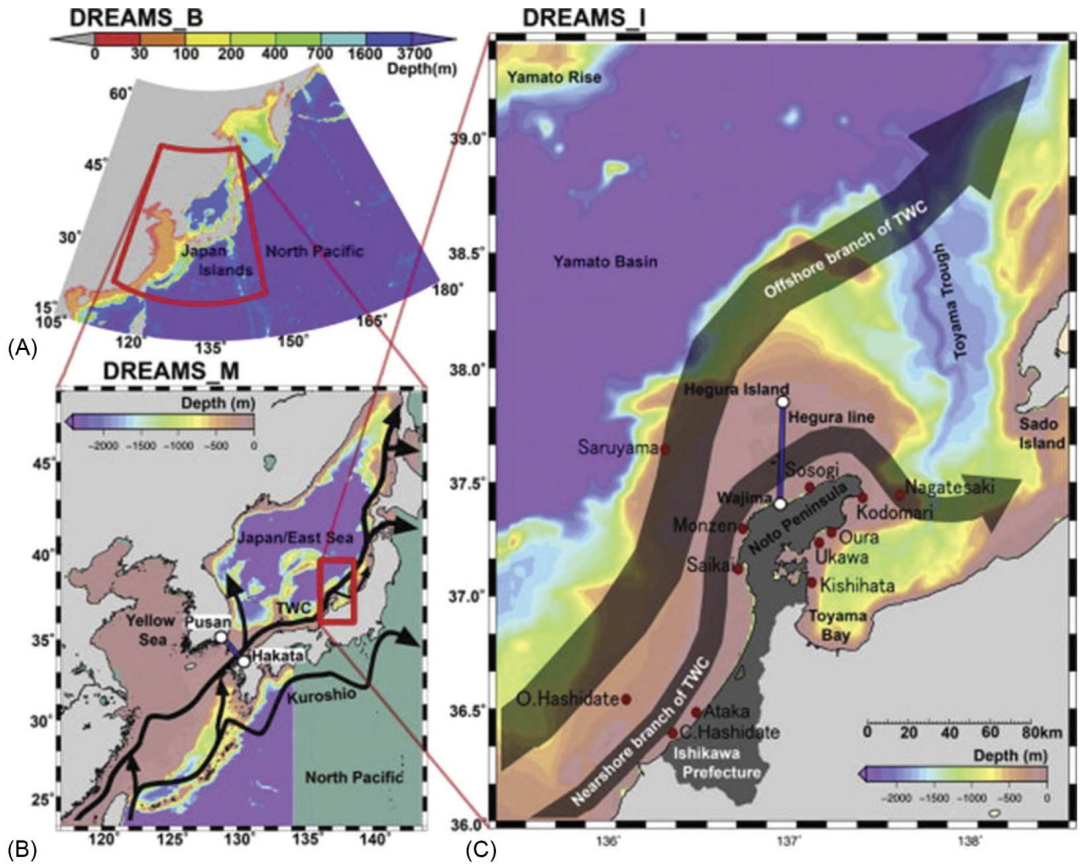


FIGURE 25.9

Copy of figure 1 from [315] of the different domains that the DREAM model can use.

transports warm waters to the northern latitudes and can have an impact on the weather as well as on the migration of mammals and fish.

Another widely used regional ocean model and four-dimensional variational data assimilation system is the Regional Ocean Modeling System (ROMS), which is an incremental strong constraint 4D VAR system. ROMS has the capability of coupled models for biochemical, biooptical, sediment, and sea ice. It has also implemented a non-Gaussian component [409]. In [192], ROMS is used as a coastal ocean model of the southern Tyrrhenian Sea. The authors were able to show that using 4D VAR, they were able to develop certain dynamical features that were not present in the free run, but could be seen in the Moderate Resolution Imaging Spectroradiometer (MODIS) image for that same location. A copy of the two assimilation runs and the MODIS image are shown in Fig. 25.10.

The ROMS system has also been used on the eastern seaboard of the United States to forecast the currents and the temperature and salinity in the New York Bight [250,498]. It is shown in [498] that the incremental 4D VAR system is able to correct mismatches of the model to the observations. As we

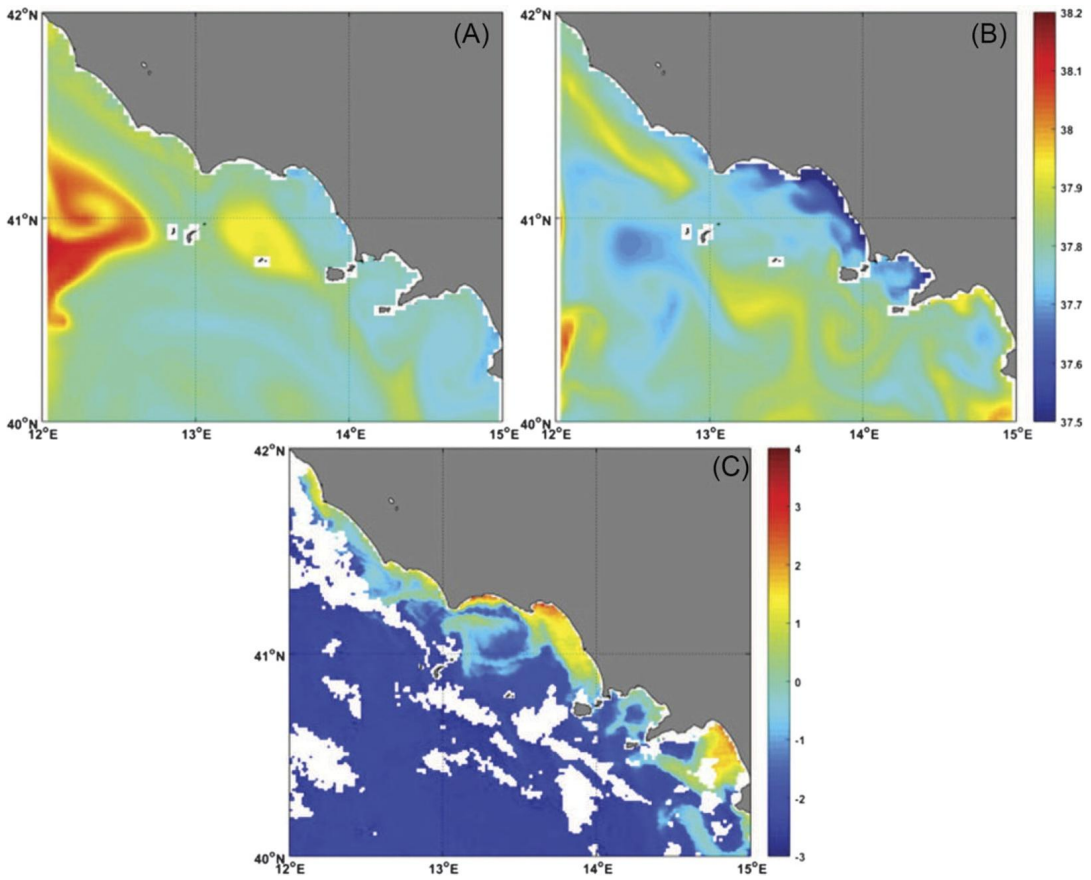


FIGURE 25.10

Copy of figure 3 from [192] of the daily salinity fields from the free run, 4D VAR compared to the MODIS image.

can see from Fig. 25.11, which is from [498], the 4D VAR system is able to correct the sea surface temperature and the current compared to the free run.

An important variable that an ocean modeling and assimilation scheme predicts is sea surface height. In [57] the authors run the US Navy's ocean model NCOM with their 4D VAR system in weak constraint form, which implies that they are approximating the model error. The 4D VAR system that is used in the Navy with their ocean model is the same as that used with the atmosphere, which means that it is observation space and is using the representer formulation.

As with the atmospheric examples, it is not only variational methods that are used in the ocean prediction community. In [278] the authors apply a particle filter for a three-dimensional biological ocean model where they are assimilating satellite observations of surface chlorophyll. They state that they are using the particle filter to avoid the use of Gaussianity and the linearity assumption of the

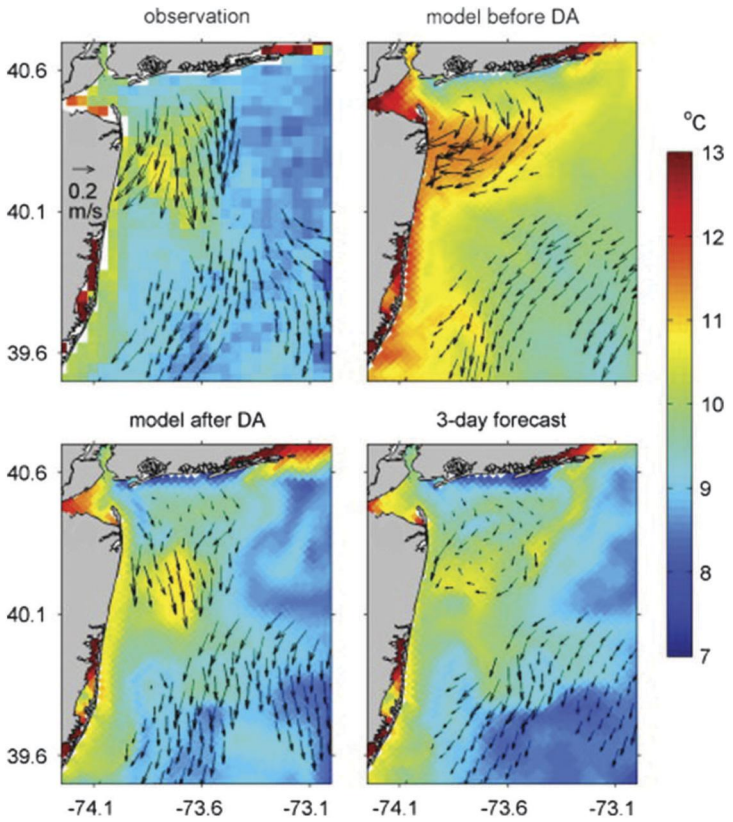


FIGURE 25.11

Copy of figure 6 from [498] of the comparison of the observed and model sea surface temperature.

EnKF. This would imply that they are analyzing a random variable that is positive definite, and cannot obtain a value below zero.

To finish off this part of the section on ocean data assimilation, we consider article [326] where the authors compare different ensemble schemes for glider data assimilation. The schemes that they compare are the EnKF and the EnOI; while we have not introduced the EnOI, the description provided in [326] is that an ensemble is run to obtain a static background error covariance matrix that is not changed throughout the assimilation period, and the updates to the analysis state are through the OI equations. The authors also introduce the FTW-EnOI; FTW stands for the *floating temporal window*, which extracts ensemble members from previous model output states. It is shown that all three assimilation methods are able to provide better fits to mooring data compared to the free run.

25.5.3 Sea Ice Data Assimilation

There has been a lot of work recently on predicting sea ice thickness and sea ice extent in Canada, where they have introduced a global ice ocean prediction system called the Global Ice Ocean Prediction

System (GIOPS). This combines a multivariate ocean data assimilation system with a 3D VAR system that produces ice analyses from assimilating satellite observations from SSM/I and SSMIS together with manual analyses from the Canadian Ice Service [400]. The authors of [400] show that it is possible to have a reduction in the spatial distribution of the RMS forecast error with the 3D VAR system than with a persistence forecast. Fig. 25.12 shows their plot of the forecast RMS for the Arctic and Antarctic regions.

We can clearly see in Fig. 25.12 that the 3D VAR system has quite a significant reduction of the RMS on the western side of the Arctic as well as throughout the upper maritime part of Canada. This is also true in the Antarctic region, where it appears that the 3D VAR system is better able to resolve a current on the outskirts of the ice extent compared to the persistence forecast.

The Canadians are also working on a regional version of an ice prediction system; however, they are developing an ensemble-variational (EnVAR) system with a 3D VAR cost function [393]. This regional ice prediction system is the Regional Ice Prediction System. This system assimilates sea ice concentration retrievals from passive microwave sensors SSM/I and SSMIS data, observations from the Advanced Scatterometer backscatter anisotropy and open water retrievals, as well as sea ice concentration from the Canadian Ice Service daily ice charts, image analyses, and lake ice bulletins. Note that these are produced manually by analysts.

Fig. 25.13 shows a copy of figure 7 from [393], and we can see that there is more agreement with the new ensemble-based assimilation system compared to the static 3D VAR-based system. We see that the new system is better able to predict the solid ice, while some differences show up in the static version.

25.6 Hydrological Applications

We shall consider three applications of data assimilation associated with hydrological process: soil moisture and groundwater, snow water equivalent, and streamflow and water storage.

Soil moisture plays a key role in the transport of goods or vehicles over non-solid surfaces, i.e., sand, mud, soils, etc. From a military point of view you do not want a transport vehicle, or a tank, becoming stuck in the mud of sand. In [156] the authors introduce the assimilation of the Gravity Recovery And Climate Experiment (GRACE) observation of terrestrial water storage (TWS); while this data is monthly and is roughly 150,000 km^2 at the equator, it contains information about vertically integrated water storage components over land, including soil moisture and groundwater. In [156] the authors state that data assimilation can be used to downscale horizontally and vertically partition the GRACE TWS observations.

In [156] the authors use the EnKF to perform the analysis, which they do for each day that they produce an analysis from the EnKF for the level 3 GRACE product, which is at one degree resolution.

The conclusion from [156] is that the assimilation of the GRACE-TWS partitioned the vertically integrated observations of column water into various water storage compartments, which are the surface and root-zone moisture, groundwater, and snow. The authors showed that groundwater storage was affected the most significantly by the assimilation of GRACE-TWS, while there was not much effect on the surface and root-zone moisture. The authors recommend that in future, assimilation should be carried out with both GRACE-TWS and SMOS or SMAP observations to improve the estimation of the complete vertical storage of moisture.

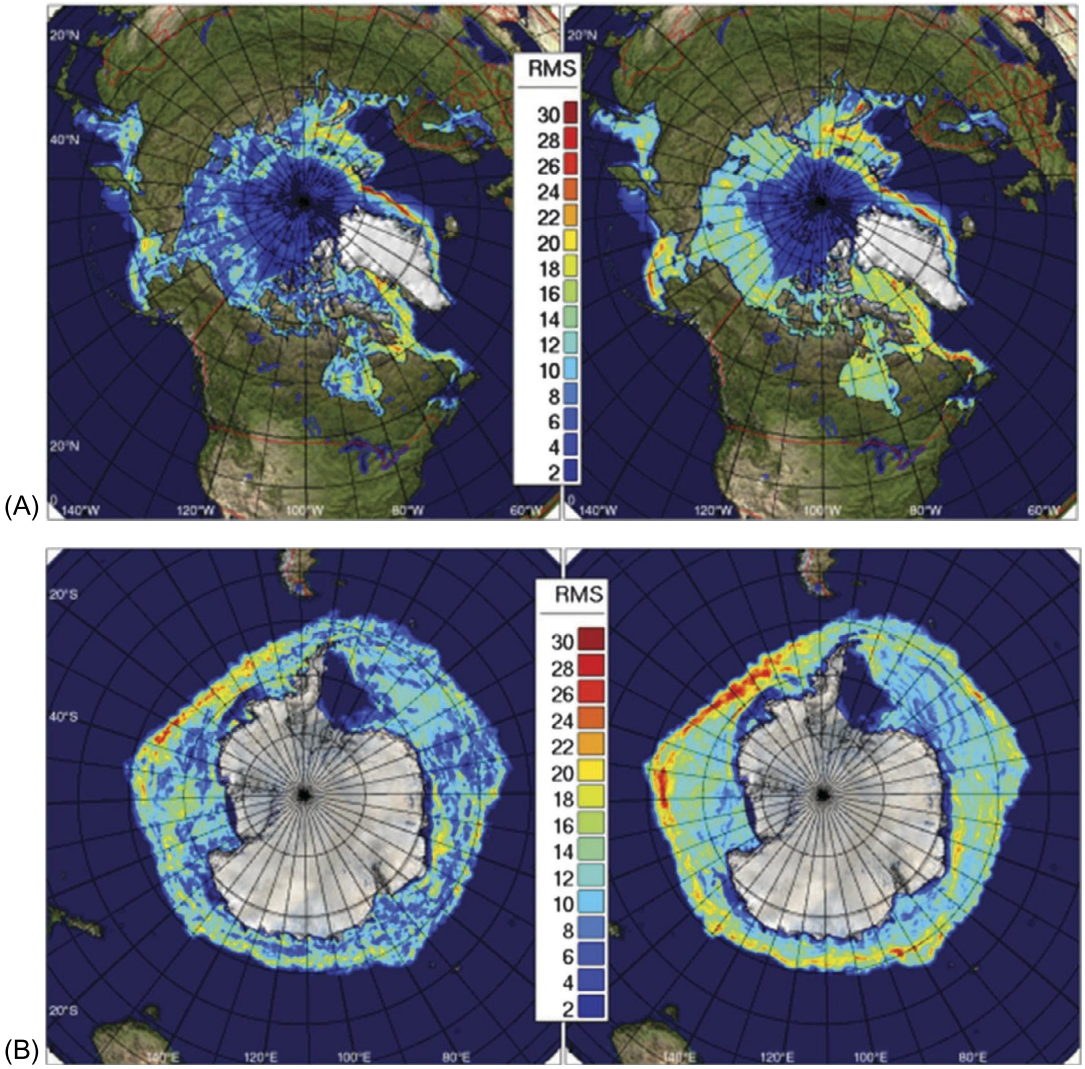


FIGURE 25.12

Copy of figures 4 and 5 of the spatial distribution of RMS errors for the data assimilation scheme and persistence from Smith, G.C., Roy, F., Reszka, M., Surcel Colan, D., He, Z., Deacu, D., Belanger, J.-M., Skachko, S., Liu, Y., Dupont, F., Lemieux, J.-F., Beaudoin, C., Tranchant, B., Drévillon, M., Garric, G., Testut, C.-E., Lellouche, J.-M., Pellerin, P., Ritchie, H., Lu, Y., Davidson, F., Buehner, M., Caya, A. and Lajoie, M. (2016), Sea ice forecast verification in the Canadian Global Ice Ocean Prediction System. Q.J.R. Meteorol. Soc., 142: 659-671. https://doi-org.ezproxy2.library.colostate.edu/10.1002/qj.2555. https://creativecommons.org/licenses/by/4.0/.

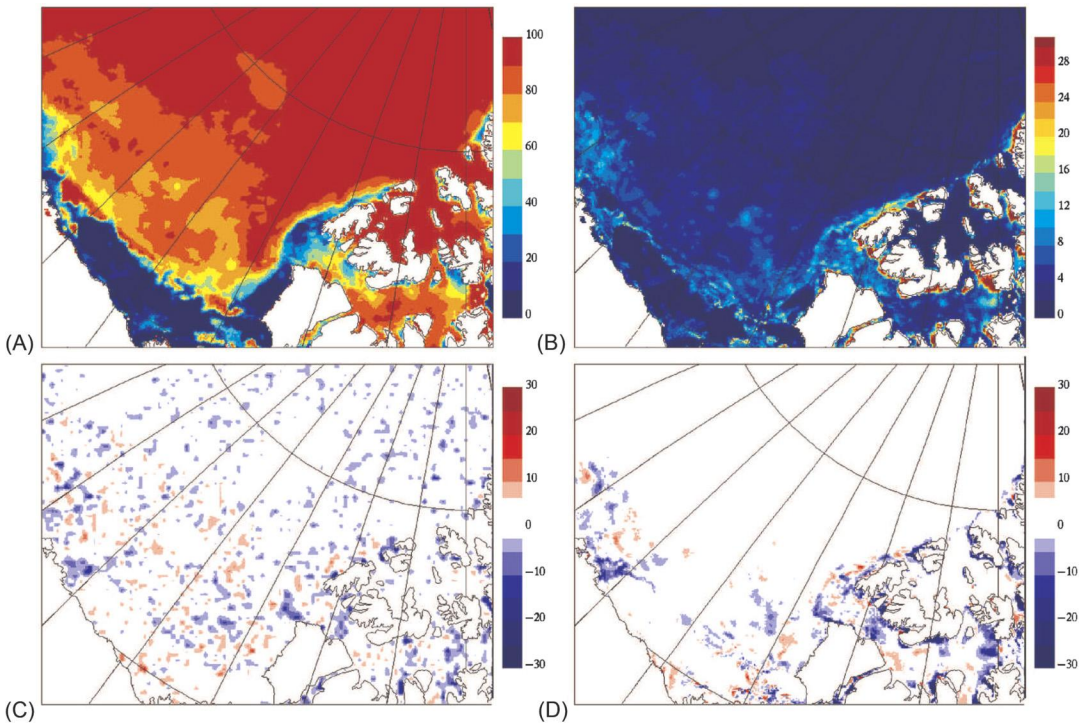


FIGURE 25.13

Copy of figure 7 from [393] of the static and ensemble background ice concentration and ensemble spread with the analysis increments from both formulations.

Since the last edition ECMWF have taken a different route and are assimilating soil moisture scatterometer data using neural networks [1]. In [1] they also consider CDF-matching, and indicate that this approach finds a simple monovariate relationship at the local scale is very dependent on the land-surface model used, where are the NNs are global multivariate models able to exploit auxiliary information and the synergy of multiple instruments, but the solution is global and no local characteristics constrain the solution. In Fig. 25.14 we have a copy of Figure 5 from [1] that illustrates the different strategies to exploit ASCAT information for soil moisture.

Snow plays a vital role in predicting the possible availability of water after a winter season. As stated in [275], water resource decisions in alpine and high altitude regions often require large-scale estimates of snow amounts and melt rates. As we mentioned earlier (Section 15.1), one may sometimes have an in situ observation of snow but wish to distribute that information over a larger domain. In [275] the authors are assimilating point snow water equivalent data with 3D VAR, EnKF, and statistical interpolation. They show that using a three-dimensional system helps to distribute the data, but that a direct insertion technique does not significantly improve the model's performance.

A problem with snow data assimilation is that snow can occur at quite high altitudes, where only certain types of remote sensing observations may be available, but if there are urban areas in the catch-

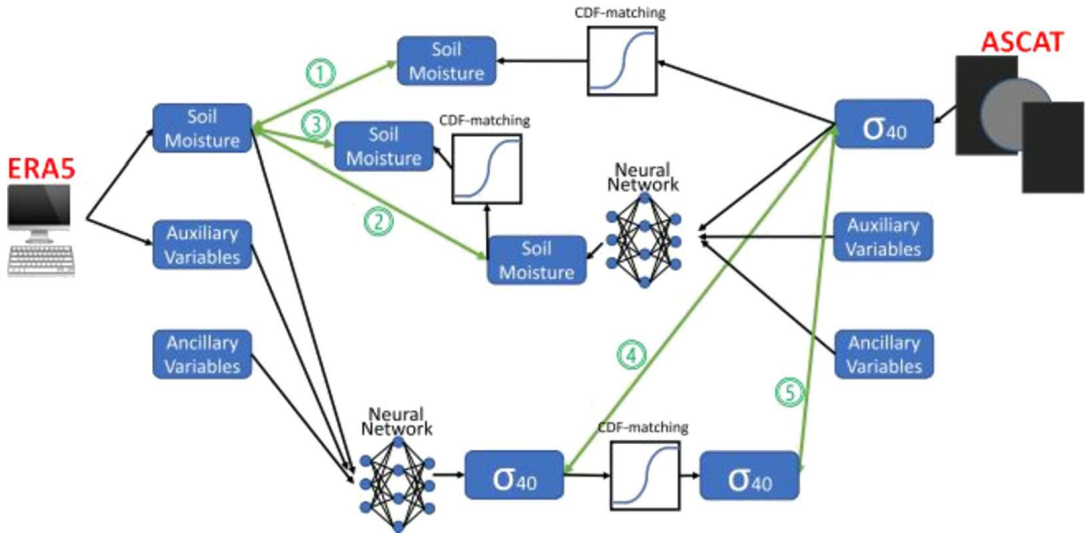


FIGURE 25.14

Copy of figure 5 from [1] of the strategies to exploit ASCAT information for soil moisture.

ment of a snow melt river, then if a sudden rapid melt occurs then we need resources to know that: (1) this is occurring and (2) how fast it is doing so. In [72] the authors develop a data assimilation system that they refer to as **particle batch smoother**, which they use to assimilate Landsat fractional snow cover area images. While in [72] the authors use this approach for retrospective snow water equivalent (SWE) estimates over several Andean study basins, it helps us to build an understanding of how snow behaves in these high-altitude regions.

Since the work in [72] a particle filter has been using to assimilate monthly snow depth observations and show that through this method they are able to improve estimations of both snow density and snow water equivalent, [402].

In [359] we have a comparative evaluation of the EKF and the maximum likelihood ensemble filter (MLEF) for real-time assimilation of streamflow data into operational hydrologic models. In this study the authors are assimilating streamflow, mean areal precipitation, and potential evaporation data for updating soil moisture. They show that due to the EKF's reliance on the linear assumption, it was not as successful in resolving the strong nonlinear relationship between assimilating streamflow to update the soil moisture. Fig. 25.15 sets out the error analysis plot to show the comparison of where the MLEF was able to resolve processes better that the EKF.

The final application that we consider here in the hydrology section is with respect to the assimilation of SWOT data for an operational reservoir management on the upper Niger river basin. In [311] the authors apply an EKF to assimilate synthetic SWOT data for the purpose of aiding the automatized control algorithm that prescribes optimal release from the Selingue dam, to benefit operational water management. Using data assimilation, the authors were able to optimize the water release from the dam such that they met the requirements for the water management where the flow from the dam was released so that they met a minimum flow requirement 300 km downstream. This is an example of the

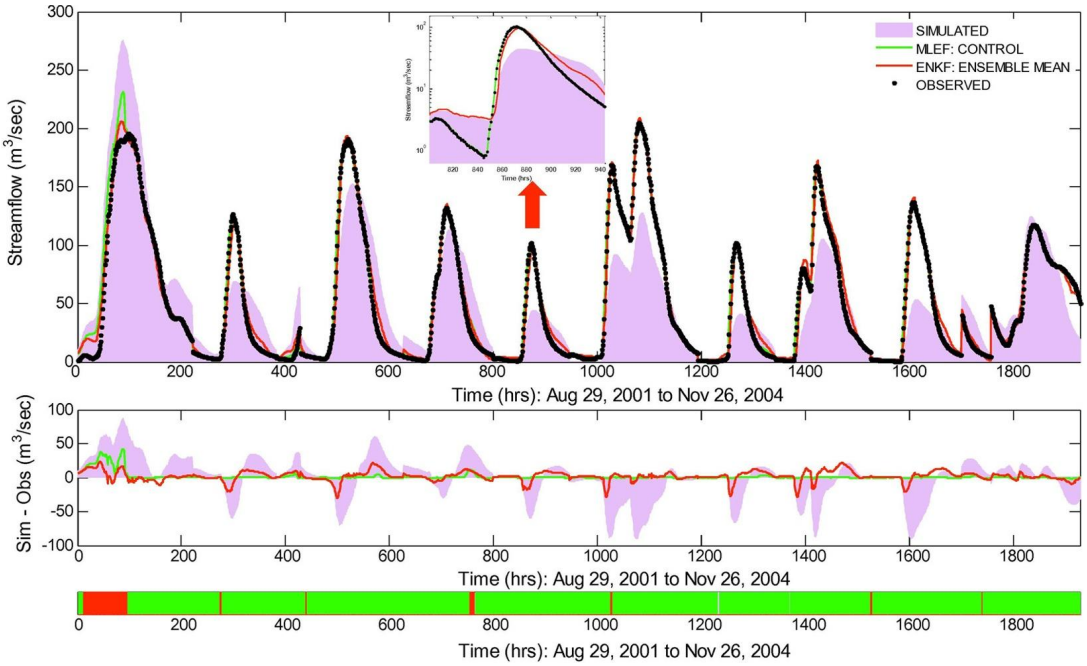


FIGURE 25.15

Copy of figure 6 from [359] of the time series of absolute error comparing the EnKF and the MLEF.

application of data assimilation that has a large impact on people's lives, which are dependent on the water from the dam at their location.

25.7 Coupled Data Assimilation

As mentioned earlier, the atmosphere interacts with the land as well as the ocean and sea ice at the surface and with the ionosphere above. In this section we shall briefly summarize work that is ongoing to find ways to couple these different components of the geophysical system together.

25.7.1 Coupled Atmosphere-Ocean Data Assimilation

A series of papers from the Department of Mathematics and Statistics at the University of Reading have developed different strategies for consideration of the coupling of the atmosphere and ocean models [143,401]. The coupling strategies are for different formulation of incremental 4D VAR systems. In [401] the authors proposed three different forms of coupled incremental 4D VAR:

- **Strongly coupled**: The control vector is for a full coupled atmospheric-oceanic numerical model. This scheme will then have a full background error covariance matrix that contains the covariances

between the atmospheric and oceanic variables. The linearization of both the observation operators and the numerical forecast model is with respect to all of fields. **Note**. There is still a lot of research into how to define the cross covariances between the atmosphere and the ocean variables. An important feature is that as this is a 4D VAR system, the background error covariance matrix is evolved through the window through the tangent linear model and the adjoint through the Hessian.

- **Uncoupled**: The two dynamical models are separate and there are two cost functions to minimize: one for the atmosphere and one for the ocean. The linearization of the nonlinear models are taken separately and there is no interaction between the two models in the outer or inner loops.
- **Weakly coupled**: This approach is a combination of the two approaches above, where the fully coupled numerical model is run for the outer loop, and its trajectory is used to linearize the atmosphere and ocean only models. Then, in the inner loops, the two separate cost functions are solved, but the two approaches interact with each other when an outer loop update is performed. This interaction comes for the SST being used as the boundary conditions for the atmosphere.

In [401] the authors assume the strong constraint approach for 4D VAR while [143] proposes strategies to account for the model error as well.

At ECMWF they have implemented a weakly coupled weak constraint of the atmosphere-ocean-wave model for the reanalysis system [234,236]. It is mentioned in [236] that the coupling system improves the ocean-atmospheric temperature estimates. In [236] the authors assess the implications of assimilating scatterometer observations. The hope of coupled atmospheric-ocean data assimilation systems is that this can allow for the better use of near surface observations and should impact both the atmosphere and the ocean.

As we have mentioned quite a few times, the US Navy's data assimilation is in observation space; this is true of both the global atmospheric and ocean systems. Given that their data assimilation systems are in the dual formulations, a different strategy has been developed to couple their assimilations schemes. In [147] the authors introduce what they refer to as the interface solver, where they assume that observations of the free atmosphere do not have a significant effect on the deep ocean. The motivation in [147] is then to say that only observations in the boundary layer between the two fluids need to be modeled. In the interface solver, the two data assimilation systems run in parallel, but they both use a coupled observation vector. We recommend reading [147] for the mathematical justification of this approach, as it does show promising signs for their dual formulation versions of data assimilation.

25.7.2 Coupled Land and Atmosphere Data Assimilation

The atmosphere does not just interact with the ocean, it also interacts with the land surface. In [382] the authors investigate the impact of soil moisture assimilation on land surface model spinup and coupled land-atmosphere prediction. In this study they are assimilating AMSR-E soil moisture retrievals with an EnKF for the land surface component. The atmospheric component is referred to as NU-WRF, which is the NASA Unified WRF model. In [382] the authors assess the impact of land data assimilation on the soil moisture and soil temperature initial conditions for WRF, land-atmosphere coupling characteristics, and ambient weather of the couple Land Information System and WRF simulations. In [382] the authors also show the impact of the soil moisture assimilation on the land surface model states and fluxes, and that depending on the strength of land data assimilation at the initialization time, significant modifications to the soil moisture flux, planetary boundary layer, and the ambient weather process chain are observed.

The land and the atmosphere are coupled in the study in [382] through what the authors refer to as land-atmospheric coupling matrices; however, these matrices are not defined in [382].

25.7.3 Coupled Atmosphere-Land-Ocean-Sea Ice Data Assimilation

The Met. Office has developed a coupled (atmosphere-land)-(ocean-sea) ice data assimilation system [246]. The coupling for the system in [246] is of the weak form, whereby the coupled model is used to provide background information for separate ocean-sea ice and atmosphere-land analyses. The increments generated from these separate analyses are then added back into the coupled model.

In [246] the authors state in the abstract that the analyses for the atmosphere and ocean are quite similar in the coupled format compared to the uncoupled, where we should note here that this system is actually the coupling of two coupled systems. However, in [246] the authors state that a positive byproduct of this experiment is that it identified two areas of the geophysical models that needed to be improved: the diurnal sea surface temperature variation and the river runoff.

25.8 Reanalysis

Reanalysis, or, as it is sometimes called, retrospective analysis, is a technique whereby observations taken over a long time period are combined objectively with a model forecast to form a time series of fields representing the state of the system [398]. The generation of the analysis fields in a reanalysis comes from a data assimilation system of some sorts. The main difference from weather forecasting is that the data assimilation system and the numerical model are kept the same over the whole time period that the time series are being generated for [398]. Therefore, given a set time period, as an example the Japanese have a 55-year reanalysis time series for the atmosphere, which is called the Japanese 55-year Reanalysis of JRA-55 [222].

Recently ECMWF completed a 110-year reanalysis product called the ECMWF Reanalysis of the twentieth century (ERC-20C: 1900-2010); this product only assimilated surface pressure and marine wind observations [340]. ERA-20C is ECMWF's first atmospheric reanalysis specifically designed for climate applications. A big difference between ERA-20C and other ERA products is that for the most past it assimilates observations that have not previously been used for numerical weather prediction.

Reanalyses are not restricted just to atmospheric data assimilation. In [276] the authors present a snow reanalysis data set for the length of the Landsat-era, over the Sierra Nevada mountain range, which covers 1985–2015. The data set in [276] is of snow water equivalent, which is based upon the assimilation of remotely fractional snow-covered area data through what they refer to as a *fully Bayesian* data assimilation method. The data assimilation method is the particle batch smoother, which generates an ensemble posterior SWE estimate, based on a prior estimate from a land surface model with a snow depletion curve. The observations of fractional snow cover are obtain as retrieved products from Landsat 5 Thematic Mapper, Landsat 7 Enhanced Thematic Mapper, and Landsat 8 Operational Land Imager reflectance data.

In [148] the authors present an in-situ based reanalysis of the global ocean temperature and salinity with the In situ Analysis System (ISAS) which is an optimal interpolation data assimilation system. ISAS produces gridded fields of temperature and salinity that preserve, as much as possible, the time and space sampling capabilities of the Argo network of profiling floats.

25.9 Ionospheric Data Assimilation

According to [151], the Earth's ionosphere is a highly dynamic region that is almost constantly in a state of flux. Solar radiation, geomagnetic activity, chemical reactions, and natural dynamics all act to perturb the state of the ionosphere. The time scales involved with the ionosphere range from hours to days. In [151] the authors introduce the Global Assimilation of Ionospheric Measurement-Gauss-Markov (GAIM-GM). The assimilation scheme in GAIM-MV is the Kalman filter, and they show in [151] that through these scheme they are able to correct ionospheric propagation delays, which helps improve geolocation and communications.

We can see the impact that the assimilation has had over the control model run of the Ionospheric Forecasting Model in the copy of figure 7 from [151] in Fig. 25.16.

Another example of ionosphere data assimilation can be found in [408], where the authors use the Kalman filter to assimilate pseudorange and carrier phases. They show that the Kalman filter approach, when applied to the strong geomagnetic event that took place on September 26, 2011, produced solutions that were within 2–4 total electron contents units of the observed values.

25.10 Renewable Energy Data Application

As we mentioned in the introduction to this book, renewable energy companies require good estimates of meteorological conditions for wind and solar plants, while tidal and river turbines require good estimates of the water flow as well as river and sea heights.

In [7] the authors compare the EnKF and static 3D VAR over a verification domain in Texas and Oklahoma for a month-long experiment with 24-hour forecasts being produced. In the experiment they wanted to determine the day-to-day performance, as well as over 10 individual wind ramp cases of the two mentioned data assimilation schemes. The authors showed that the flow dependency helped the EnKF to produce better forecast throughout the verification window, but that the averaging in the EnKF degraded the analysis for the wind ramp event, suggesting that a best member technique to analyze could be better for these cases.

This is an important piece of work with respect to wind power companies, which are required to shut down their turbines if the wind strength exceeds a certain threshold. There have been several cases of exploding wind turbines because they were not shut down before a wind ramp event.

Fig. 25.17 shows a copy of figure 5 from [7], where we can see that the 3D VAR is producing the isotropic and homogeneous increments associated with the static formulation, while for the EnKF we can see the different structures in the increment, which are more in tune with the possible dynamical flow of the errors.

25.11 Earthquakes

This section comprises of two parts: Optimal interpolation and what is referred to as **Green's function** assimilation applied to tsunami and long-period ground motions.

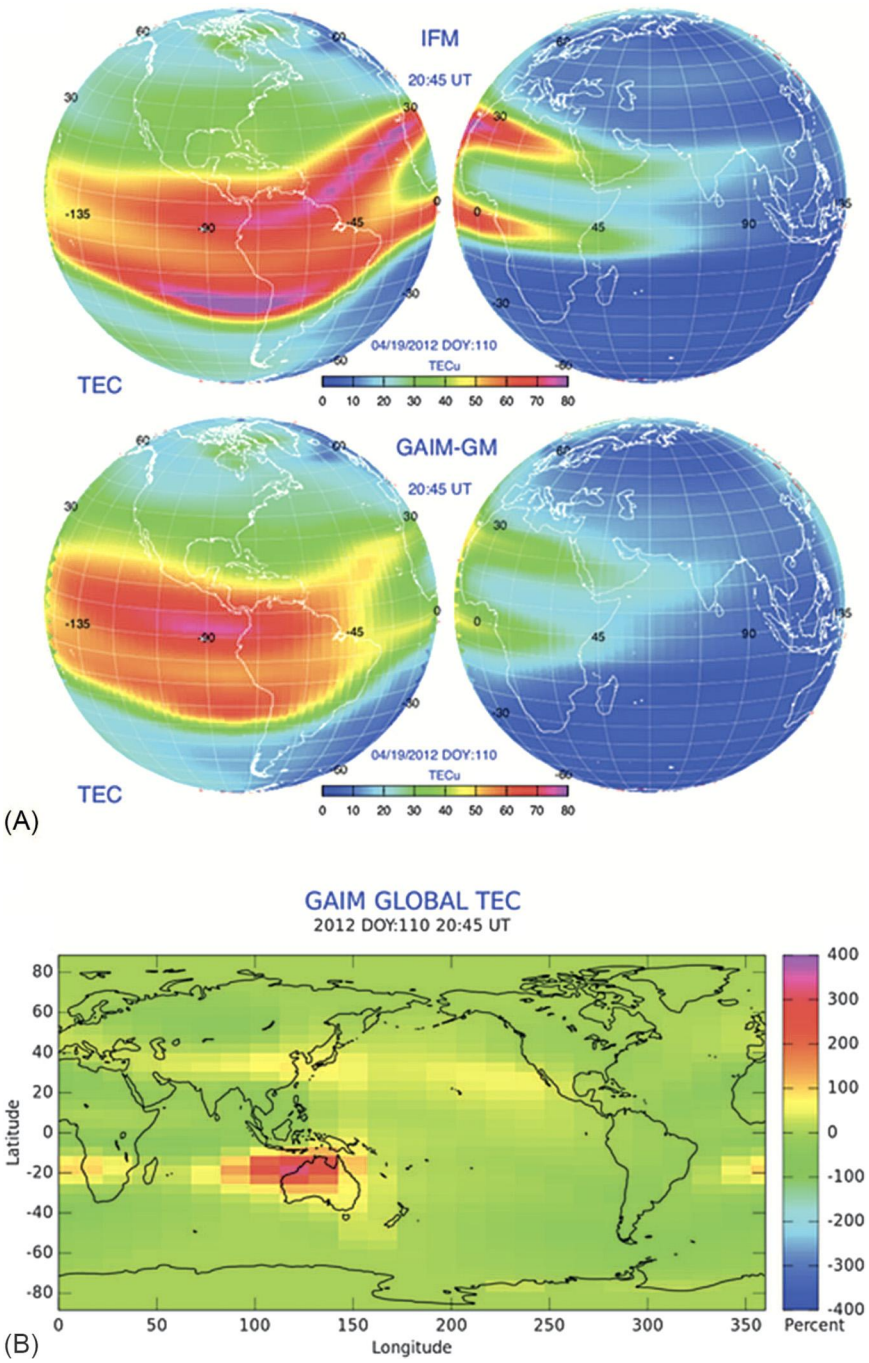


FIGURE 25.16

Copy of figure 7 from [151] of the TEC for with and without data assimilation.

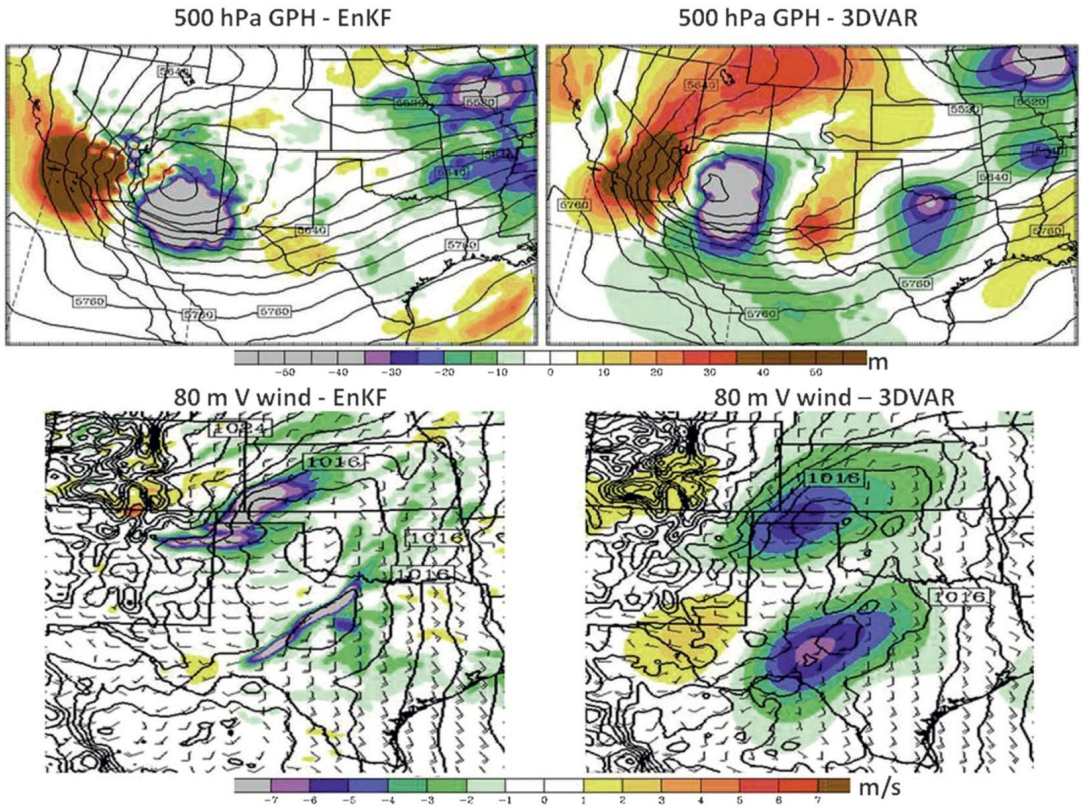


FIGURE 25.17

Plots of the background and analysis increments for the EnKF and 3D VAR from Ancell, B. C., Kashawlic, E., and Schroeder, J. L. (2015). Evaluation of Wind Forecasts and Observation Impacts from Variational and Ensemble Data Assimilation for Wind Energy Applications, Monthly Weather Review, 143(8), 3230-3245.

25.11.1 Optimal Interpolation

Starting with tsunami assimilation from [466] they start with assuming that the total grid number is L, and the total observations is m. The tsunami wavefield at the n-th time step is represented by

$$\mathbf{x}_n = \left(h\left(n\Delta t, x, y\right), M\left(n\Delta t, x, y\right), N\left(n\Delta t, x, y\right)\right)^T,$$

where h is tsunami and M and N are velocities in the horizontal directions. The data assimilation consists of two steps: a propagation step and an assimilation step. The propagation step is expressed as

$$\mathbf{x}_n^f = \mathbf{F}\mathbf{x}_{n-1}^a, \tag{25.6}$$

where $\mathbf{F}$ is the tsunami propagation matrix, such that given the tsunami's wavefield at time $t = (n-1)\,\Delta t$, the foretasted tsunami wavefield at the next time step $t = n\Delta t$ is calculated, usually a 2-D linear long-wave tsunami model is adopted.

In the assimilation step we have

$$\mathbf{x}_n^a = \mathbf{x}_n^f + \mathbf{W}\left(\mathbf{y}_n - \mathbf{H}\mathbf{x}_n^f\right). \tag{25.7}$$

The observation operator is a large and sparse matrix that extracts the foretasted tsunami height at the m stations, where the weight matrix $\mathbf{W}$ is the solution of the linear system

$$\mathbf{W}\left(\mathbf{R} + \mathbf{H}\mathbf{P}^f\mathbf{H}^T\right) = \mathbf{P}^f\mathbf{H}^T. \tag{25.8}$$

By iterating between the propagation and assimilation steps, the tsunami wavefield is assimilated. During or after the assimilation process, it is possible to obtain the forecasted tsunami waveform at a point of interest (PoI) based on the forward tsunami simulation using the present assimilated tsunami wavefield as the initial condition

For the purpose of early warnings, the tsunami waveforms at the points of interest need to be forecasted. The assimilated tsunami wavefield can be used in tsunami forward modeling to forecast the tsunami at the points of interest. This can be done progressively whenever a new assimilated wavefield is generated. However, the tsunami wavefield in the entire modeling domain needs to be repeatedly calculated during the assimilation process, this is shown in Figure 1a from [466], and we have a copy of this figure in Fig. 25.18.

One approach to deal with this is to update the tsunami forecast every 10 minutes and a new wavefield is created at intervals of 1 s. As a result of high computational costs the adoption of other more complex but more realistic models such as the linear dispersive tsunami model is not practical, which may limit the accuracy of the tsunami forecast in cases where the tsunami dispersive character is evident.

If we now consider [323] they are looking at long-period (LP) ground motions with periods of $\sim 2-10$ s caused by large earthquakes that are strongly amplified in sedimentary basins, posing serious threat to modern cities to cause resonance and damage to skyscrapers, oil storage tanks, long-span bridges, and other structures with long natural periods.

In [323] they state that since the LP ground motions are composed of surface waves traveling for longer distances with much slower speeds than body waves, an alert could be issued by detecting the spread of strong ground motion near the source and before large ground motions occur in distant basins. In [323] the present an early forecasting of LP ground motions based on data assimilation of observed ground motions obtained by a high-density, nationwide seismic networks and a computer simulation of the seismic wave propagation using a high-resolution subsurface structure model.

In [323] they have the problem set up in the same way as [466] but they explained that $\mathbf{W}$ is calculated so that the covariance between the assimilated and the actual wavefield is minimized, the weight at the j-th station with respect to the g-th grid, w_{gj}, is calculated by solving the equation:

$$\sum_{j=1}^{M} w_{gj}\left(\mu_{ij}^b + \delta_{ij}\rho_i\rho_j\right) = \mu_{gi}^b, \tag{25.9}$$

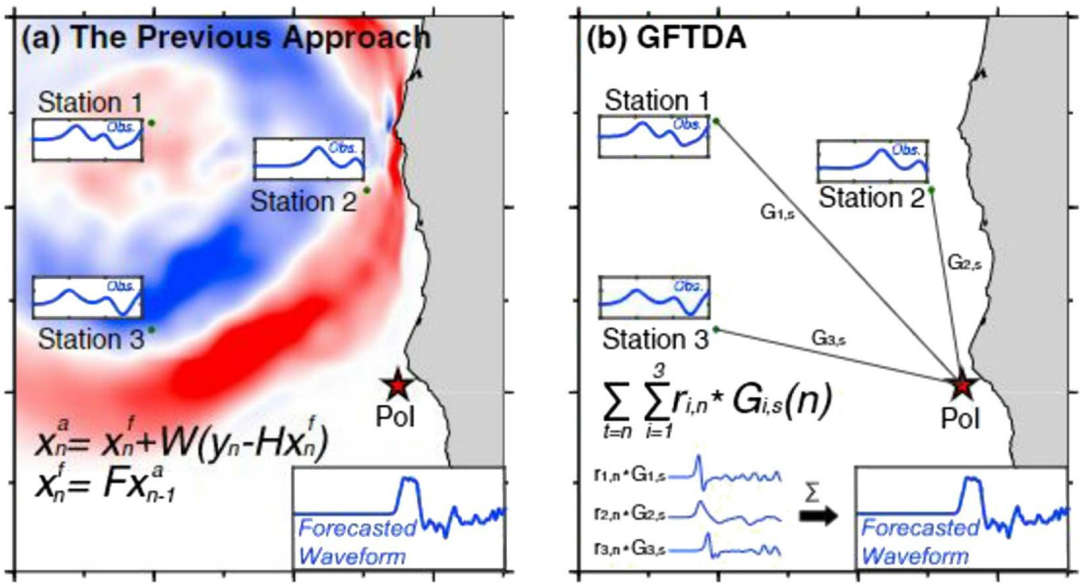


FIGURE 25.18

Copy of Figure 1 from [466].

by considering the correlation of the errors between the forecasted wavefield at each pair of the numerical grid point and that of station $\left(\mu^b_{ij(gi)}; i = 1, 2, \ldots, M\right)$ and the ratio of the error of the observed waveform (σ_o) relative to that of the forecasted waveform (σ_b) at the ith station $\rho_i = \frac{\sigma_o}{\sigma_b}$.

As most of the observing station are at the surface, data assimilation can only be performed in the model at the surface, but in [323] they state that the construction of the 3-D wavefield from the assimilated wavefield on the surface proceeds steadily as the surface wave propagates for a distance of about a wavelength.

In Fig. 25.19 we have a copy of figure 2 from [323] that is an illustration of how the OI approach works in the situation just described. In Fig. 25.20 we have a copy of figure 4 from [323] that shows the snapshots of the assimilated wavefield at 30, 60, 90, and 120 s from the earthquake occurrence time, obtained by the data assimilation between the observed ground motions at the stations and the simulated wavefield.

A statement that is made in [323] to motivate to consider the Green's function approach is that modern HPC with efficient parallel computing capabilities enable faster forecast simulations of seismic waves at speeds faster than the actual speed of the surface waves.

25.11.2 Greens Function Data Assimilation

Returning to [466] to address the problems identified with the OI approach, Wang et al. introduce the Green's Function-based Tsunami Data Assimilation (GFTDA). Here Green's functions are used in data assimilation for tsunami forecasting. In the optimal interpolation approach, if the residual between the

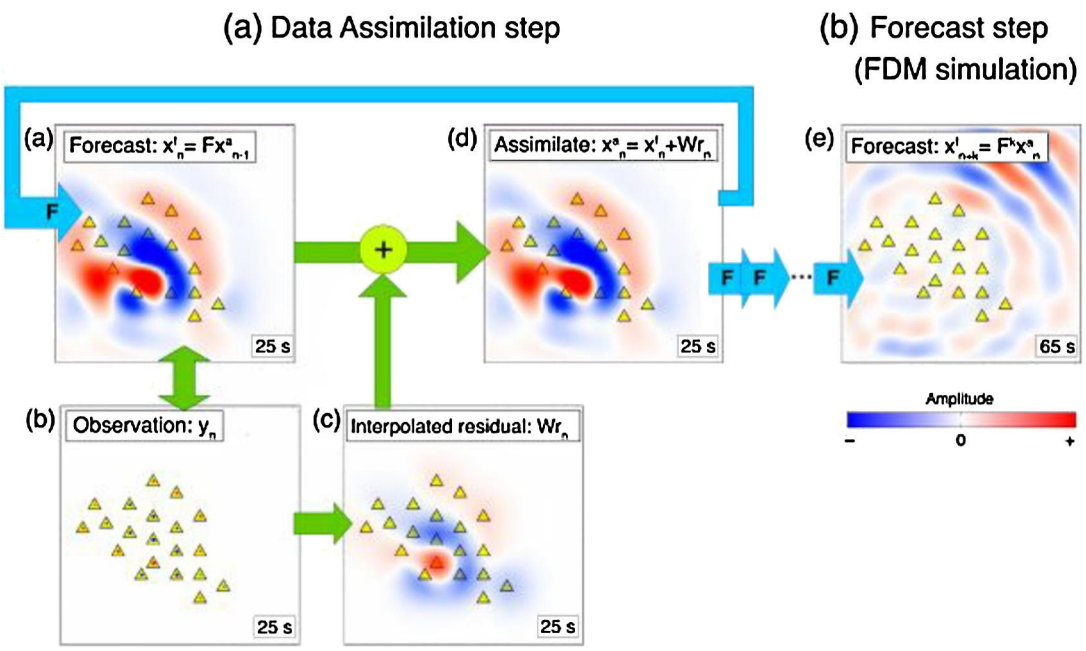


FIGURE 25.19

Copy of figure 2 from [323].

observed and forecasted tsunami height is nonzero, the station will have an assimilation response, which can be near Gaussian shape if the distance between stations are larger than the characteristic distance of 10 km of matrix $\mathbf{P}^f$ and $\mathbf{R}$.

If we now define the residual vector as $\mathbf{x}_n^r \equiv \mathbf{W}\left(\mathbf{y}_n - \mathbf{H}\mathbf{x}_n^f\right)$, that represents the residual of each station at time $t = (n-1)\,\Delta t$, then this can be rewritten as a linear combination of the unit column vector $\mathbf{e}_i^T$ multiplied by the corresponding residual of tsunami height at i-th station r^i, given by

$$\mathbf{x}_n^r = \mathbf{W}\sum_i r_n^i \mathbf{e}_i^T. \tag{25.10}$$

The assimilation response will propagate across the region under consideration, following the tsunami propagation model. This will result in changes in the tsunami height and velocity at other grid points. When the tsunami wave propagation model is linear, the assimilation response of different steps and different stations can be superposed. Therefore, let the waveform at the j-th grid point resulting from the propagation of the i-th station's assimilation response with the Green's function, $G_{i,j}$. We can see this in the right hand side plot in Fig. 25.18.

Storing the output of Green's functions requires computer memory, but the amount of memory required can be reduced by limiting the number of PoIs. For a given region with a tsunami observation network and PoIs, the number of Green's functions to be calculated is $m\,(N+m)$, where m is the number of observation stations and N is the total number of PoIs. The PoIs should include nearshore points

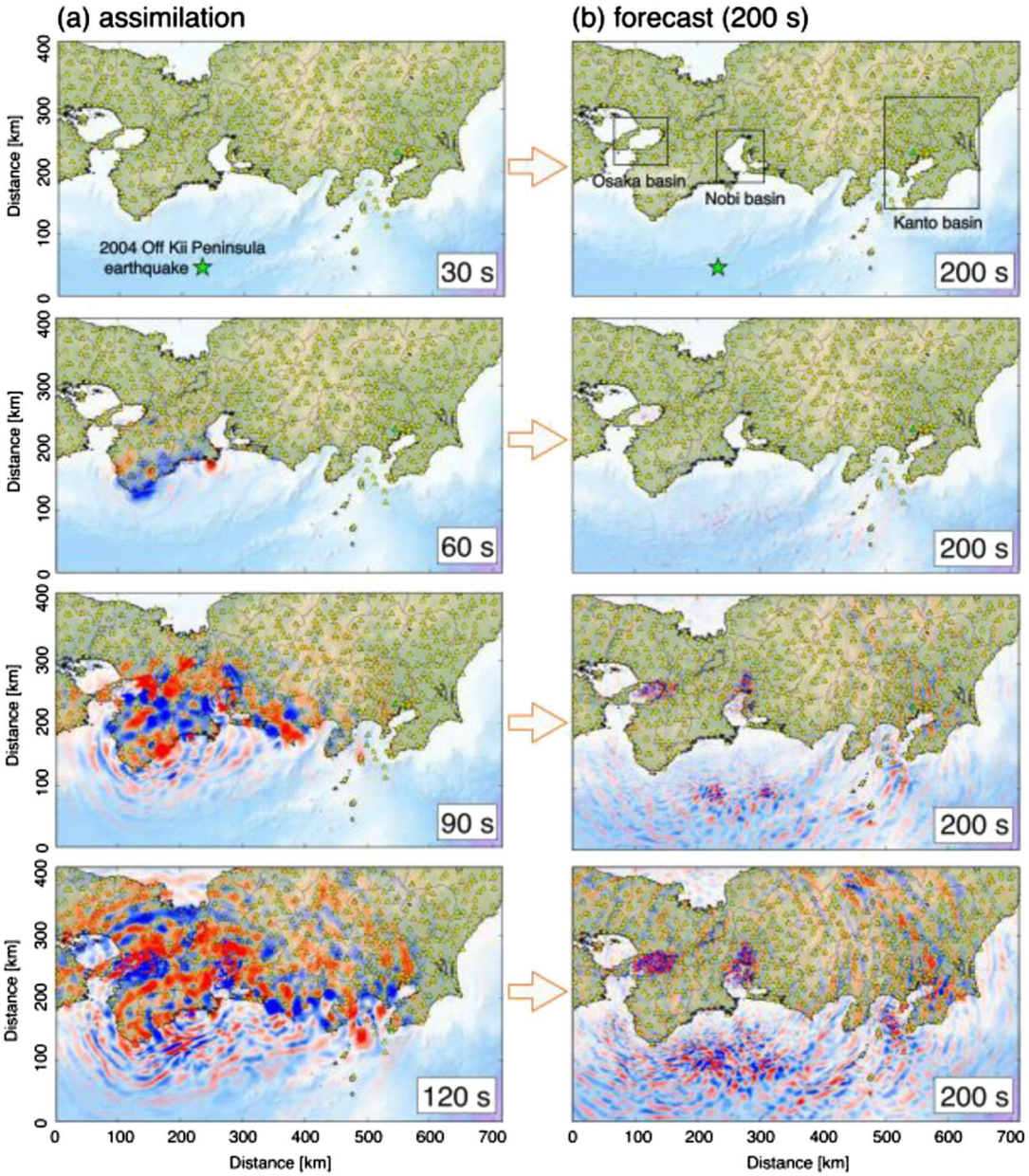


FIGURE 25.20

Copy of figure 4 from [323].

because forecasting the tsunami height and velocity at these points would be of greatest importance to the residents living nearby. Conversely, forecasting the tsunami at points that are far away from the coast would not be useful for the coastal communities. Therefore, the computation of Green's functions at points far away from the coast can be omitted to obtain a manageable number of Green's functions.

In Fig. 25.21 we have a copy of figure 3 from [466] showing the results from an experiment using the GFTDA approach that shows the observed waveforms at 11 PoIs nearshore stations indicating some interesting results.

In [323] they provide a more detailed mathematical derivation for the Green's function approach which we proved here. They start by saying that Green's function is formally defined as the response of displacement wavefield corresponding to the single force input; however, in [323] they use the term as the wave propagation response to a unit wavefield input, e.g. acceleration or velocity input at the data assimilation station.

From the propagation equation we have that

$$\mathbf{x}_n^a = F\mathbf{x}_{n-1}^a + \mathbf{W}\mathbf{r}_n = \mathbf{F}^n\left(\mathbf{W}\mathbf{r}_0\right) + \mathbf{F}^{n-1}\left(\mathbf{W}\mathbf{r}_1\right) + \cdots + \mathbf{W}\mathbf{r}_n = \sum_{i=0}^{n} \mathbf{F}^{n-1}\left(\mathbf{W}\mathbf{r}_i\right), \tag{25.11}$$

where $\mathbf{r}_0 = \mathbf{y}_0$ as the forecasted wavefield is zero at the beginning.

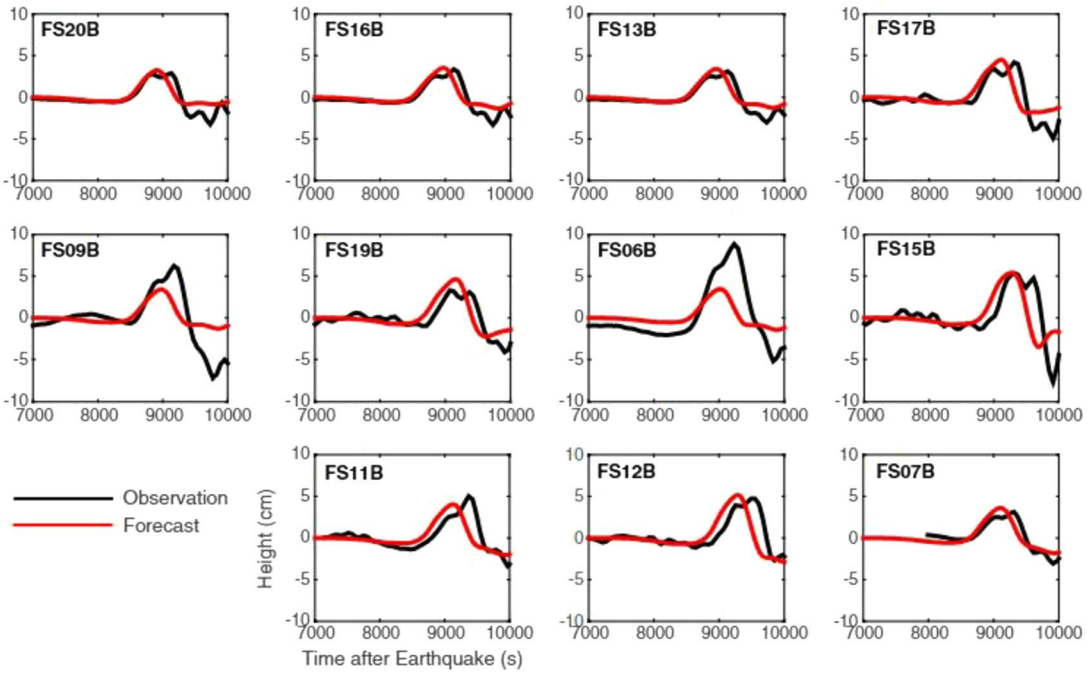


FIGURE 25.21

Copy of figure 3 from [466].

Next the residual matrix at time $t = 1, 2, \ldots, n$ is defined as $\boldsymbol{\Upsilon}_t$, using the residual vectors $\mathbf{r}_t^i$ at each station, i, and a unit vector, $\mathbf{e}_i$, as

$$\boldsymbol{\Upsilon}_t = \sum_{i=1}^{M} \mathbf{e}_i \left(\mathbf{r}_t^i\right)^T . \tag{25.12}$$

Thus the analysis state can be written as

$$\mathbf{x}_n^a = \sum_{t=0}^{n} \mathbf{F}^{n-1} \left(\mathbf{W} \sum_{i=1}^{M} \mathbf{e}_i \left(\mathbf{r}_t^i\right)^T \right) = \sum_{i=1}^{M} \sum_{t=0}^{n} \mathbf{F}^{n-t} \mathbf{W} \mathbf{e}_i \left(\mathbf{r}_y^i\right)^T . \tag{25.13}$$

Therefore, the forecast of the future wavefield after k time steps by numerical simulation, can be obtained similarly by the convolution of the residual at each data assimilation station, $\mathbf{e}_i \left(\mathbf{r}_t^i\right)^T$, and the Green's function of the wave propagation response, $\mathbf{F}^t$, from station i to j, multiplied by the weight matrix, $\mathbf{W}$, as

$$\mathbf{x}_{n+k}^f = \sum_{i=1}^{M} \sum_{t=0}^{n} \mathbf{F}^{n+k-1} \mathbf{W} \mathbf{e}_i \left(\mathbf{r}_t^i\right)^T , \tag{25.14}$$

where $\mathbf{F}^{n+k-1}\mathbf{W}\mathbf{e}_i$ is the weighted Green's function, that describes the wave propagation response from i to j, multiplied by a spatially distributed weight of the optimum interpolation operator.

The weighted Green's functions are calculated in advance by simulations, so that forecast of ground motions at the target site can be achieved instantaneously by convolving them with the residual at each data assimilation station instead of running expensive 3-D FDM simulations in real time.

The Green's function from data assimilation stations i to the forecast target sites j is calculated by loading a single unit wavefield given as the delta source time function at station i for each component $l = x, y, z$ (Fig. 25.22a). The weighted Green's function can be obtained by summing up a set of Green's functions corresponding to the unit wavefield input on grids g around station i with weight W_{gi}, (Fig. 25.22b).

Here, the force vector, $\mathbf{We}_i^t$, of the weighted unit wavefield around i for component l is

$$\mathbf{We}_i^t = \left(\; w_{1i}, \quad W_{2i}, \quad \cdots \quad , W_{gi} \quad \cdots , W_{Gi} \; \right). \tag{25.15}$$

As the Green's function of the wave propagation response $\mathbf{F}$ has the components $\mathbf{F}_{jg}\left(x, y, z, t; x', y', z', t'\right)$, denoted as $\mathbf{F}_{jg}^{tm}$, for $m = x, y, z$ which represents the response of component m and point j corresponding to the unit wavefield input of component l at point g. The component m of the weighted Green's functions $\left(\mathbf{FWe}_i^l\right)_j^m$ is the superposition of the response, $\mathbf{F}_{jg}^{lm}\mathbf{W}_{gi}$, for a set of weighted unit wavefield inputs, Fig. 25.22b, as:

$$\left(\mathbf{FWe}_i^l\right)_j^m = F_{j1}^{lm} W_{1i} + F_{j2}^{lm} W_{2i} + \cdots + F_{jg}^{lm} W_{gi} + \cdots + F_{jG}^{lm} W_{Gi} . \tag{25.16}$$

As it is quite often that case that the total number of forecast target sites is much smaller than the stations used for data assimilation; it is then possible to use the reciprocity theorem to exchange g, j and l, m; thus

$$\left(\mathbf{FWe}_i^l\right)_j^m = F_{1j}^{lm} W_{1i} + F_{j2}^{lm} W_{2i} + \cdots + F_{gj}^{lm} W_{gi} + \cdots + F_{Gj}^{lm} W_{Gi} . \tag{25.17}$$

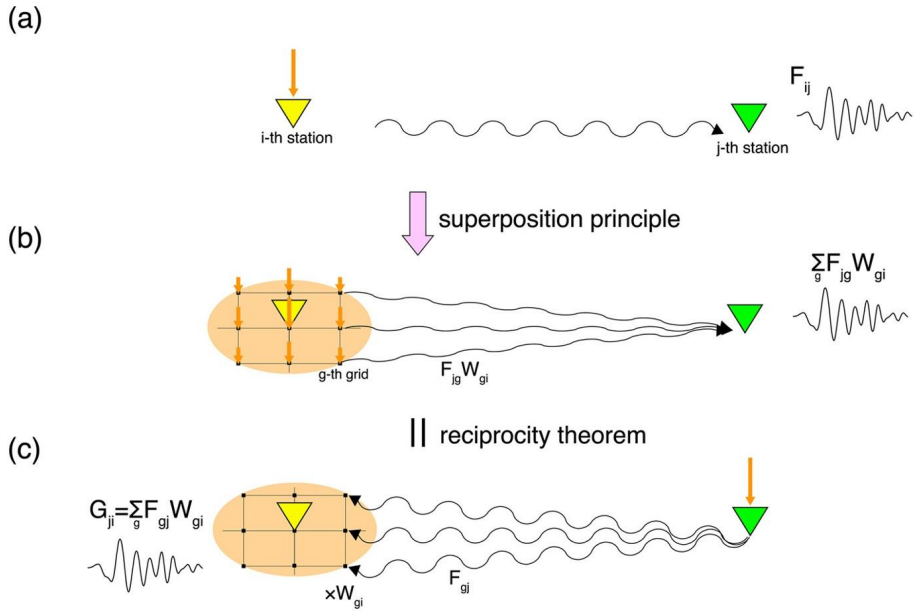


FIGURE 25.22

Copy of figure 7 from [323].

The weighted Green's function can be then obtained very efficiently with a unit wavefield input on the forecast target site $j \ll i$ to obtain waveforms at the grid points of the simulation around the data assimilation station i, and then summing them up with the weights W_{gi}, Fig. 25.22c.

In Fig. 25.23 we have a copy of figure 8 from [323] that show the comparison of the forecasted waveforms of the north-south component of ground velocity with the observation from a ground station during the 2004 off Kii Peninsula earthquake.

25.12 Oil and Natural Gas

One application of data assimilation in this section, as with all data assimilation systems, is to attempt to produce better forecasts of the permeabilities of porous media for oil, gas, and water flows as well. A second application we show is for oil reservoir characteristics. In both cases an ensemble-based approach is used.

In [197] the authors show that it is feasible, and physically plausible, to estimate absolute and relative permeabilities jointly under multiphase flow conditions, where this refers to the flow of two or three phases, i.e., oil-water or gas-oil, through assimilating measured historical data, combined with a prior model. They show that before data assimilation there was a large variability in the relative permeability parameters; after data assimilation the parameters became closer to their reference values,

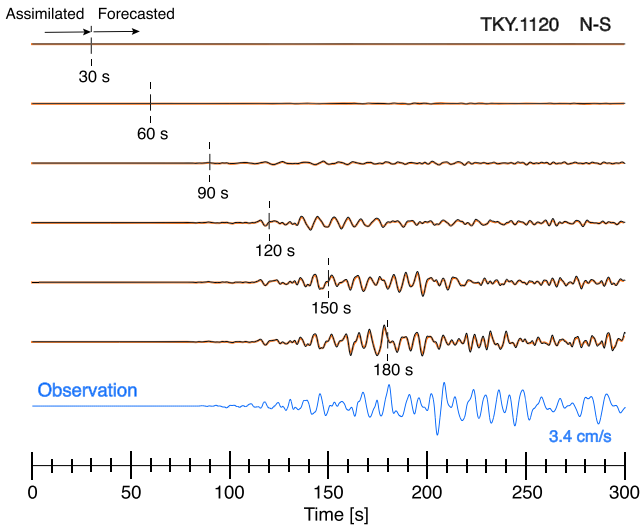


FIGURE 25.23

Copy of figure 8 from [323].

but also the variability of the updated parameters was reduced. They were also able to show that there was a reduction in the RMSE of the log-permeability versus time.

In [247] the authors introduce a new version of an ensemble Kalman smoother where they use cluster covariance and compare its performance against the EnKF and the EnS for the characterization of the oil reservoirs. This is important as it is required for determining operational schedules, and the use of infill drilling. An important finding in [247] is the fact that the EnKF and the EnS do not perform well when the permeability is bimodal, while their cluster covariance approach would have ensemble members clustering at both modes. As such, the authors of [247] can analyze these modes correctly, and show this in Fig. 25.24.

25.13 Biogeoscience Application of Data Assimilation

In this section we have applications of data assimilation to improve prediction of crop yields of different varieties as well as for the optimization of parameters in a carbon and hydrology dynamic ecosystem model.

In [185] the authors present experiments with a full field 4D VAR system to show an improvement in winter wheat yield estimation through assimilating leaf area index (LAI) from Landsat TM and MODIS data. Winter wheat comprises 85% of China's total summer grain production. This implies that the need for accurate regional monitoring of wheat growth and yield predictions have become crucial for national food security and sustainable agriculture development in China. An important finding from the work in [185] was that when the authors applied a correction to the Landsat retrieval, they were able to obtain more accurate results, but this indicates that the accuracy of the retrieval needs to be improved

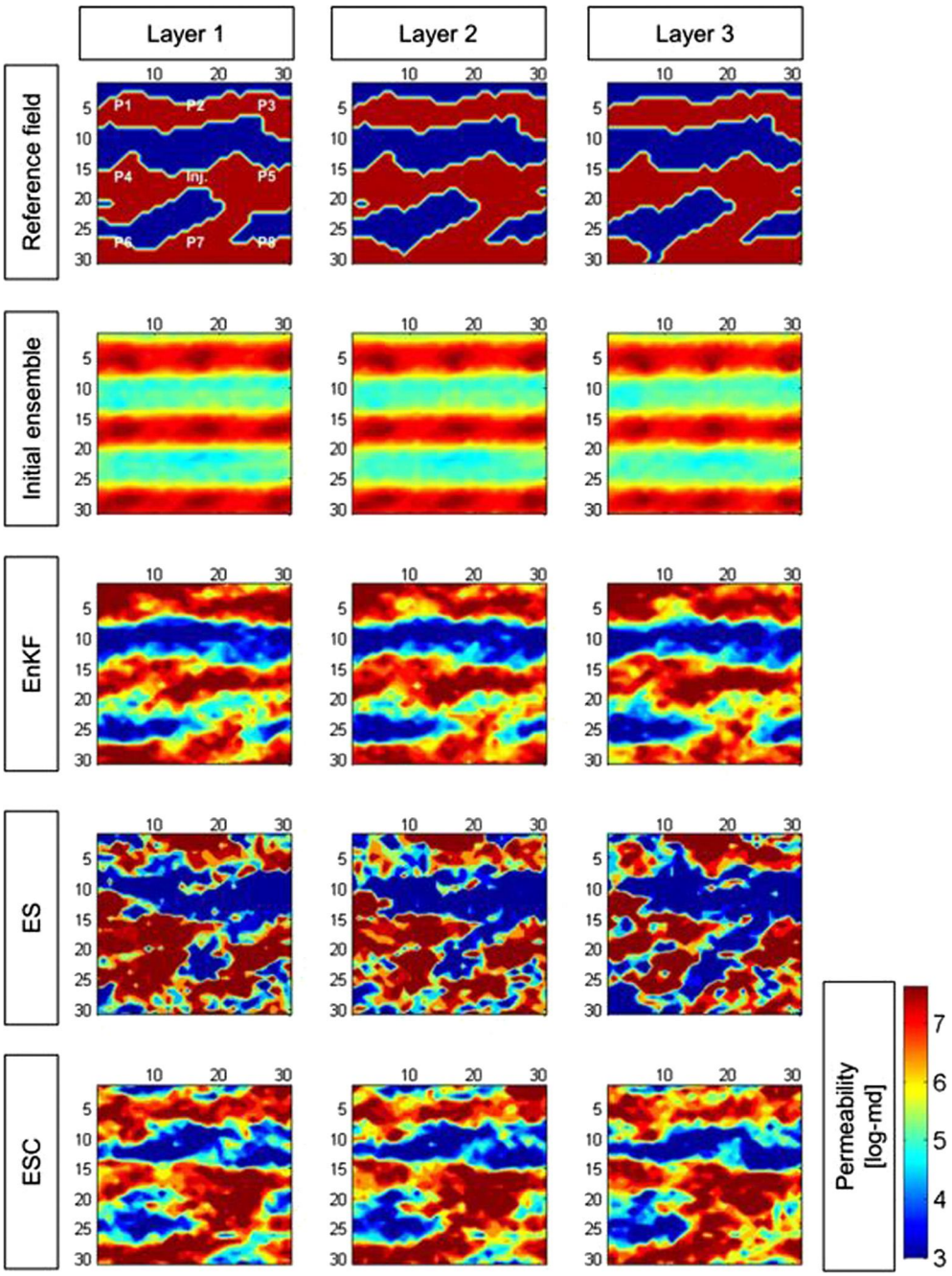


FIGURE 25.24

Copy of figure 7 from [247] of the mean updated log-permeability fields.

during the different phenological stages of the crop. They also show that the assimilation of the MODIS current 1 km LAI product was not suitable for their assimilation system. We have a copy of figure 12 from [185] (Fig. 25.25) to show the difference between the control run and the two assimilation runs, indicating that data assimilation can improve the crop yields for this specific crop in their setup.

The LAI is a vital observation when it comes to crop yield estimation. As we just saw, it is important for the estimation of the winter wheat yields in China, while in [199] the authors again use a full field 4D VAR system to assimilate MODIS LAI data, but this time for the estimation of maize. The 4D VAR system was designed to estimate temporally continuous LAI at the regional scale. In [199] the authors were able not only to show that the assimilation technique was able to improve the accuracy of the LAI, but also to capture features of continuity and evolutional information of LAI.

In [19] the authors apply a 3D VAR scheme to be able to assimilate jointly eddy covariance flux measurements and fraction of photosynthetically active radiation products over temperate forest to optimize the parameters in a process-oriented biosphere model. This work is important as it highlights the use of data assimilation to improve the parameters is biosphere model so that the uncertainties in the parameters are reduced, such that forecast of changes to the biosphere due to changing environments and levels of carbon, but also to be able to match current values, are more reliable.

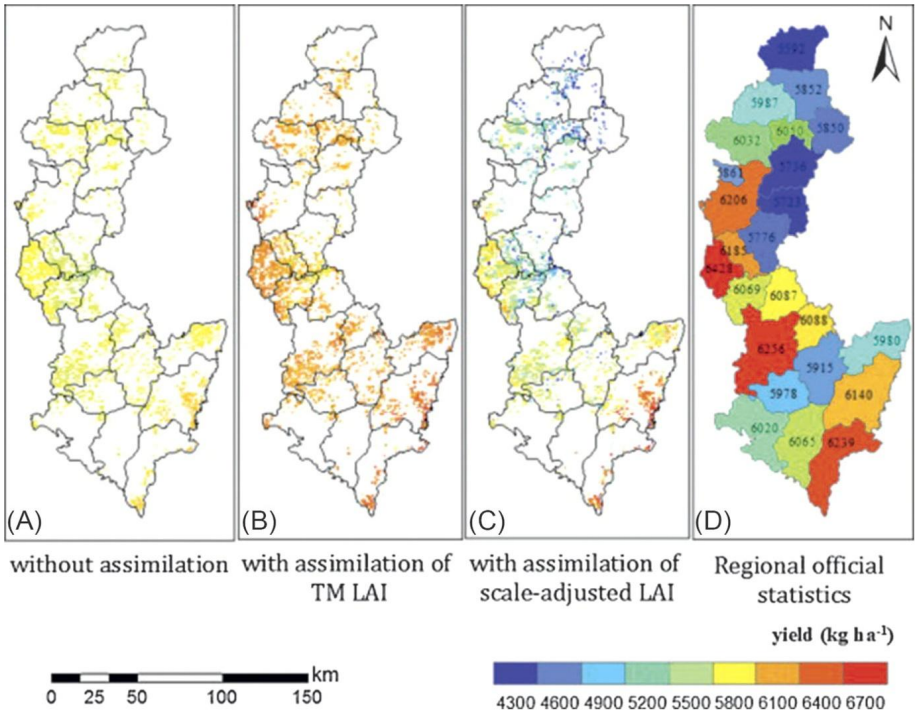


FIGURE 25.25

Copy of figure 12 from [185] of the estimated winter wheat yield with and without data assimilation.

25.14 Other Applications of Data Assimilation

In this section we have a list of a few other applications of a form of data assimilation outside of the geophysical disciplines that we have presented in this chapter so far:

- **Tsunami prediction**. Optimal interpolation of tsunami waveforms from seafloor pressure gauges [162].
- **Atmospheric inversion of urban carbon dioxide**. 4DDA (3D VAR with temporal nudging) with the coupled WRF and WRF-CHEM models [238].
- **Carbon dioxide inversion at multiple scales over highly inventoried agricultural landscape**. A comparison of the EKF and two different versions of 3D VAR [390], where they determined the feasibility of CO_2 inversion methods to produce regional carbon flux estimates, capturing regional CO_2 flux estimates at subnational scales.
- **Mars atmospheric analysis**. 4D LETKF assessing the impact of window lengths on the diurnal features of Mars [500].
- **Paleoclimate**. Reconstructing past climate nudging, particle filter, EnSRF [296].
- **Reconstructing turbulent flow around aircraft**. ETKF [209].
- **Stratospheric chemistry**. Ensemble fix-lag Kalman smoother [298].
- **Thermal records from borehole in the oceanic crust**. MCMC [481].
- **Olivine single crystals**. MCMC [310].
- **Crust and upper mantle structures**. MCMC [329].
- **Thermospheric modeling through solar cycles 23 and 24**. EnOI [312].
- **Magnetosphere data assimilation of low-altitude magnetic perturbations**. OI [294].
- **Gas storage reservoirs**. Assimilation of surface displacements to improve geomechanical parameters of gas storage reservoirs—ensemble smoother [502].
- **Traffic-Flow models** - Assimilating Eulerian and Lagrangian data using either EnKF or particle filters [483].

25.15 Summary

In this, our last chapter of this book, we have presented different applications of data assimilation in its many forms for a few of the geoscience disciplines. We have seen that data assimilation can be used to predict large-scale synoptic weather features down to trying to improve tornado track and intensity forecasts. In this chapter we have introduced the concept of correcting for position errors by introducing an augmented control vector for the displacement parameters.

We have seen how data assimilation is used to help optimize fishing around the coast of Japan, as well as modeling the current correctly in the New York Bight. We have seen that data assimilation is used to model coupled ocean-biology systems as well as for the prediction of surface winds.

In the hydrological fields, we have seen that data assimilation is used to help optimize the flow of water out of a dam to meet environmental conditions 10,000 km downstream, as well as improving models of groundwater storage, soil moisture at the surface as well as in the root zone. We have seen that data assimilation can be used to model, and predict, snow cover, and snow water equivalency in high-altitude zones where there are not many in situ observations, but then only providing point source

data, yet data assimilation can still spread that point data around to nearby points to help improve the simulations.

Sea ice in the Northern Canadian maritime region is becoming more important to predict so that shipping there can be optimized. We saw that the Canadians have developed a global variational sea ice system, while in the regional area they have developed an ensemble-based approach. However, we have also indicated that these different geophysical systems may all need to be optimized simultaneously in some form so that the feedback between the systems is consistent. The process of consolidating the geophysical systems as we saw is referred to as coupling, and we introduced the concept of strong and weak coupling for both the primal and dual forms of 4D VAR.

In this edition we have add sections of JEDI, OSEs, and OSSEs, as well as a new section on earthquake related data assimilation that introduced Green's functions as part of the data assimilation process.

We have seen that data assimilation can be used to optimize the use of renewable energy by predicting ramp up events that can affect wind turbines, as well as for the prediction of clouds that would reduce the amount of solar power that can be generated, but that data assimilation can also be used to help optimize the production of oil and natural gas.

Data assimilation methods have been used to create time series over long periods for atmospheric variables, ocean and snow, where observations are assimilated consistently with the same data assimilation system and numerical model. These systems can be used to create relatively accurate climate data to be used either as initial conditions for other assimilation schemes, numerical models, or as verification data.

Data assimilation has also been shown to be beneficial in the prediction of the interactions of the ionosphere with income geomagnetic radiation as well as on Mars to predict diurnal features on that planet.

This brings us to the end of this book. We hope that the theory has been beneficial and that as we walked you through the different versions of data assimilation currently available, you were able to develop your own thoughts about which technique could be the best approach for your own applications, but also that you can see where a change in the theory may fit your application. We acknowledge that artificial intelligence/machine learning/deep learning and more and more being integrated in to data assimilation to improve their performance as well as their forecast errors.

Data assimilation is a continually changing field. As a Ph.D. student at the University of Reading in the early 2000s, I remember the discussions about whether or not it was beneficial to run a second outer loop at ECMWF—now they are running 50 4D VARs in parallel, and performing 3 to 5 outer loops.

There are still many different problems left for all of the forms of data assimilation; most noticeable is that the non-Gaussian problem is not going to go away. However, it will be difficult for even the variational methods to adapt to some of the distributions that we may have to find the maximum likelihood state for. Yet this is the challenge of data assimilation: to adapt to the situation that is put in front of that scheme.

This page intentionally left blank

CHAPTER

Solutions to Select Exercise 26

Chapter 2

- **Exercise 1**:

$$\mathbf{A}^T = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}, \quad \mathbf{B}^T = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \\ 3 & 6 & 10 \end{pmatrix}, \quad \mathbf{C}^T = \begin{pmatrix} 1 & -5 & 2 & 2 \\ -5 & 0 & 3 & 0 \\ 2 & 3 & -3.5 & 3 \\ 0 & 0 & 7 & 0 \end{pmatrix}. \tag{26.1}$$

  Only **B** is symmetric.
- **Exercise 3**:

$$\begin{aligned} |\mathbf{A}| &= (4-1)(-2) - (-2-1) + (1+2) \\ &= -6 + 3 + 3 \\ &= 0, \end{aligned}$$

  therefore, **A** is singular

$$\begin{aligned} |\mathbf{B}| &= 4(21-5) - 1(7-2) + (5-6) \\ &= 64 - 5 - 1 \\ &= 58, \end{aligned}$$

  therefore **B** is nonsingular.

$$\begin{aligned} |\mathbf{C}| &= 1(0-0) - 2(0-60) + 3(0-40) \\ &= 0 + 120 - 120 \\ &= 0, \end{aligned}$$

  therefore **C** is singular as well. Note we could have expanded the determinant about the last row and would have seen straightaway that **C** was singular.
- **Exercise 5**:

$$|\mathbf{A} - \lambda\mathbf{I}| = \begin{vmatrix} 1-\lambda & 2 \\ 2 & \\ & 24-\lambda \end{vmatrix} = (1-\lambda)(4-\lambda) - 4 = 0$$
$$= \lambda^2 - 5\lambda = 0 \Rightarrow \lambda_1 = 0, \lambda_2 = 5.$$

Data Assimilation for the Geosciences. https://doi.org/10.1016/B978-0-32-391720-9.00030-9

It can easily be verified that the eigenvectors for these two eigenvalues are $\mathbf{v}_1 = \begin{pmatrix} 2 \\ -1 \end{pmatrix}$ and $\mathbf{v}_2 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$.

Chapter 3

- **Exercise 1**: We need to form $\mathbb{E}\left[X^2\right] = \int_{-\infty}^{\infty} x^2 \exp\left\{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right\} dx$. Therefore, introducing the change of variable $t = \frac{x-\mu}{\sigma}$ implies that $x = \sigma t + \mu$ and that $dx = \sigma dt$. Checking the limits of integration, we have that when $x \to -\infty$ then $t \to -\infty$ and when $x \to \infty$, then $t \to \infty$ which means that the limits of integration for t as the same as x. This then implies that we have to evaluate

$$\int_{-\infty}^{\infty} \left(\sigma^2 t^2 + 2\sigma\mu t + \mu^2\right) \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{t^2}{2}\right\} dt.$$

We saw in the derivation of the Gaussian distribution that $\int_{-\infty}^{\infty} t^2 \exp\left\{-\frac{t^2}{2}\right\} dt = \sqrt{2\pi}$ and $\int_{-\infty}^{\infty} t \exp\left\{-\frac{t^2}{2}\right\} dt = 0$, using these identities and recognizing that the term multiplying μ^2 is the definition of the cumulative distribution function of the standard Gaussian distribution, then we have that $\mathbb{E}\left[X^2\right] = \sigma^2 + \mu^2$.

Chapter 5

- **Exercise 2**: We need to form Euler's equation and as such we require $F = x^2 y'^2$, therefore we have $F_{y'} = 2x^2 y'$, which leads to

$$0 - \frac{d}{dx}\left(2x^2 y_0'\right) = 0 \Rightarrow \frac{d}{dx}\left(2x^2 y_0'\right) = 0 \Rightarrow y_0' = \frac{A}{2x^2}.$$

Integrating with respect to x we obtain $y_0 = -\frac{A}{2x} + B$. Now, applying the boundary conditions $y_0(1) = 1$ and $y_0(2) = \frac{1}{2}$, we have $\left.\begin{array}{l} 1 = -\frac{A}{2} + B \\ \frac{1}{2} = -\frac{A}{4} + B \end{array}\right\} A = -2,\ B = 0$, therefore we have that $y_0 = \frac{1}{x}$ is the extremal.

For the second case the boundary conditions give the same answer of $y_0 = \frac{1}{x}$, but we have to note that we are on the interval $[-1, 1]$ which includes the origin and as such this function is not defined there, let alone in $\mathbb{C}^2$. Therefore, this is no extremal.

- **Exercise 4**: $F \equiv y^2 + y'^2 - 2y\sin x$, therefore we have that $F_y = 2y - 2\sin x$, $F_{y'} = 2y'$, which implies that the Euler equation for this situation is

$$2y - 2\sin x - \frac{d}{dx}(2y') = 0 \Rightarrow y'' - y' = \sin x.$$

Therefore the complimentary function can easily be shown to be $y_{cf} = Ae^x + b^{-x}$. For the particular integral we try $y_{pi} = \alpha' x + \beta' \cos x$, which upon substituting y_{pi} into the second-order differential equation above, and applying the boundary conditions, yields the general expression for the ex-

tremum as

$$y_0(x) = Ae^x + Be^{-x} + \frac{1}{2}\sin x.$$

- **Exercise 5**: $F \equiv \frac{\sqrt{1+y'^2}}{y}$ and $F_{y'} = \frac{y'}{y\sqrt{1+y'^2}}$. The first integral, because the functional is independent of x, is given by

$$F - y'F_{y'} = C \Rightarrow \frac{\sqrt{1+y'^2}}{y} - y'\frac{y'}{y\sqrt{1+y'^2}} = C.$$

After cancelations and some rearranging, we obtain

$$\left(\frac{dy}{dx}\right)^2 = \frac{1-C^2y^2}{C^2y^2} \Rightarrow \frac{dy}{dx} = \pm\frac{\sqrt{1-C^2y^2}}{Cy} \Rightarrow \pm\int\frac{Cydy}{\sqrt{1-C^2y^2}} = \int dx + A.$$

Integrating directly, we have that $\mp\frac{1}{C}\sqrt{1-C^2y^2} = x + A$, therefore squaring both sides yields $\frac{1}{C^2}\left(1-C^2y^2\right) = (x+A)^2$. Upon substituting the boundary conditions, we obtain the solution

$$\left(x - \frac{1}{2}\right)^2 + y^2 = \frac{5}{4},$$

which is the equation for a circle centered at $\left(\frac{1}{2}, 0\right)$ and radius $\frac{\sqrt{5}}{2}$ as required.

Chapter 6

- **Exercise 1**: For the matrix $A = \begin{pmatrix} 0 & 1 \\ 0 & -2 \end{pmatrix}$ then we have the system of differential equations given by $\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -2x_2 \end{cases}$. Hence we have that $x_2(t) = Ae^{-2t}$ and $x_1(t) = -\frac{1}{2}A^{-2t} + B$. Forming the fundamental solution where we require that $\Phi_1(t_0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ implies that $A = 0$ and $B = 1$. Applying the second condition, $\Phi_2(t_0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, then we have that $A = e^{2t_0}$ and $B = \frac{1}{2}$. Therefore, combining the two solutions yields the state transition matrix as

$$\Phi(t, t_0) = \begin{pmatrix} 1 & \frac{1}{2} - \frac{1}{2}e^{2(t_0-t)} \\ 0 & e^{2(t_0-t)} \end{pmatrix}.$$

- **Exercise 2**: We are seeking to prove that using the exponential of the time invariant matrix is equivalent to solving the differential equations for the state transition matrix. We have that $\Phi(t, t_0) = \begin{pmatrix} 1 & 1 - e^{t_0-t} \\ 0e^{t_0-t} & \end{pmatrix}$. If we form the product $\mathbf{AA}$, we have

$$\mathbf{A}^2 = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 0 & 1 \end{pmatrix}.$$

It can easily be shown that $\mathbf{A}^n = \begin{pmatrix} 0 & (-1)^{n-1} \\ 0 & (-1)^n \end{pmatrix}$, which then implies that we have the definitions of the summation approximation of e^{-x} in the entries in the arrays. However, the top right entry does not have the 1 term in the summation as it is not a function of the identity matrix and as such it is $1 - e^{-x}$, which then proves that the two solutions are equivalent for this case.

- **Exercise 3**: We need to form the controllability matrix, so recognizing that $\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and that $\mathbf{B} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ then $\mathbf{AB} = \begin{pmatrix} a \\ c \end{pmatrix}$. Therefore the controllability matrix is given by $\mathbf{U} = \begin{bmatrix} \mathbf{B} & \mathbf{AB} \end{bmatrix} = \begin{pmatrix} 1 & a \\ 0 & c \end{pmatrix}$ which will only have rank 2 is $c \neq 0$. This then implies that a, b, and d are arbitrary.

Chapter 7

- **Exercise 3**: We have that $\min \int_0^1 u^2 dt$ subject to $\begin{matrix} \dot{x} = z, \\ \dot{z} = u - z \end{matrix}$, with $x(0) = z(0) = 0$, and $x(1) = \frac{1}{2}\left(e + e^{-1}\right) - 1$ and $z(1) = \frac{1}{2}\left(e - e^{-1}\right)$.
Forming the Hamiltonian, we have

$$H = -u^2 + \lambda_1 x + \lambda_2 (u - z).$$

Therefore, we have

$$\begin{aligned} \dot{\lambda}_1 &= -\frac{\partial H}{\partial x} = 0 \Rightarrow \lambda_1 = A, \\ \dot{\lambda}_2 &= -\frac{\partial H}{\partial z} = -\lambda_1 + \lambda_2 \Rightarrow \lambda_2 = Be^t + A, \\ 0 &= \frac{\partial H}{\partial u} = -2u + \lambda_2 \Rightarrow u = \frac{\lambda_2}{2} = \frac{1}{2}\left(Be^t + A\right). \end{aligned}$$

We now have to solve the coupled differential equations

$$\begin{aligned} \dot{x} &= z, \\ \dot{z} &= -z + \frac{1}{2}\left(Be^t + A\right). \end{aligned}$$

It can easily be shown through applying the technique of integrating factors, combined with the initial and end time conditions, that

$$\begin{aligned} x^* &= \frac{1}{2}\left(e^t + e^{-t}\right) - 1, \\ z^* &= \frac{1}{2}\left(e^t - e^{-t}\right), \\ u^* &= e^t, \end{aligned}$$

and that the functional at the minimum is $J^* = \int_0^1 e^{2t} = \frac{1}{2}\left(e^{2t} - 1\right)$.

Chapter 8

- **Exercise 1**: $C_0 = \sum_{i=0}^{k} \alpha_i = 1 - \frac{3}{2} + \frac{1}{2} = 0$ and $C_1 = \sum_{i=0}^{k} i\alpha_i - \beta_i = -\frac{3}{2} + 2 - \frac{5}{4} + \frac{3}{4} = 0$, which implies that the method is consistent. For the zero stability we have $Z^2 - \frac{3Z}{2} + \frac{1}{2} \Rightarrow Z_0 = 1$, and $Z_1 = \frac{1}{2}$ which implies zero stability. The bound on the truncation error can be shown to be $\frac{53}{24} |M_3| h^2$, where M_3 is the bound on the third-order derivative. Therefore,

$$\mathbf{A} = \begin{pmatrix} 0 & 1 \\ -\frac{1}{2} & \frac{3}{2} \end{pmatrix}, \quad \mathbf{B}_n = \begin{pmatrix} 0 & 0 \\ -\frac{3}{4}Ln & \frac{5}{4}L_{n+1} \end{pmatrix}.$$

Next we have $\mathbf{e}_{n+1} = \begin{pmatrix} 0 & 1 \\ -\frac{1}{2} - \frac{3hL_n}{2} & \frac{3}{2} + \frac{5hL_{n+1}}{2} \end{pmatrix} \begin{pmatrix} e_n \\ e_{n+1} \end{pmatrix} + \begin{pmatrix} 0 \\ h\tau_n \end{pmatrix}$. We require the eigenvalues of $\mathbf{A}$, which can be shown to be $\lambda = 1, \frac{1}{2}$, such that the bound on the norm of $\mathbf{A}^n$ can be shown to be $\|\mathbf{A}^n\| \leq 9$. Therefore, from the convergence theorem we have

$$|e_n| \leq \frac{9}{2} t_n e^{18Lt_n} \frac{53}{24} h^2 M_3,$$

where $L = \max\{L_n L_{n+1}\}$.

Chapter 9

- Discretizing with a central difference scheme yields

$$L_h(y_i) = -\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + e^{x_i} \frac{(y_{i+1} - y_{i-1})}{2h} + \sin(x_i)\, y_i = 0.$$

The general equation is given by

$$\left(-1 - \frac{e^{x_i}}{2}h\right) y_{i-1} + \left(2 + h^2 \sin x_i\right) y_i + \left(-1 + \frac{e^{x_i}}{2}h\right) y_{i+1} = 0.$$

The first equation is given by

$$\left(2 + h^2 \sin h\right) y_1 + \left(-1 + \frac{e^h}{2}\right) y_2 = 0.$$

The last equation is given by

$$\left(-1 - \frac{e^{(N-1)h}h}{2}\right) y_{N-1} + \left(2 + h^2 \sin(N-1)h\right) y_N = 1 - \frac{e^{(N-1)}}{2}h.$$

The bound on the truncation error can be shown to be

$$|\tau_i| \leq \frac{h^2}{12}\left(M_4 + 2e^{\pi} M_3\right).$$

To ensure that the tridiagonal system to be diagonally dominant to ensure that $\mathbf{A}^{-1}$ exists, we need $h \leq \frac{2}{e^{\pi}}$.

This page intentionally left blank

Bibliography

[1] F. Aires, P. Weston, P. de Rosnay, D. Fiarbairn, Statistical approaches to assimilate ASCAT soil moisture information - I. Methodologies and first assessment, Q. J. R. Meteor. Soc. 147 (2021) 1823–1852.

[2] H. Akima, A new method of interpolation and smooth curve fitting based on local procedures, J. Assoc. Comput. Mach. 17 (1970) 589–602.

[3] M.A. Alaka, R.C. Elvander, Optimum interpolation from observations of mixed quality, Mon. Wea. Rev. 100 (1972) 612–624.

[4] J.T. Ambadan, Y. Tang, Sigma-point Kalman filter data assimilation methods for strongly nonlinear systems, Mon. Wea. Rev. 66 (2009) 261–285.

[5] J. Amezcua, P.J. Van Leeuwen, Gaussian anamorphosis in the analysis step of the EnKF: a joint state-variable/observation approach, Tellus A 66 (2014) 23493.

[6] B.C. Ancell, G.J. Hakism, Comparing adjoint- and ensemble-sensitivity analysis with applications to observation targeting, Mon. Wea. Rev. 135 (2007) 4117–4134.

[7] B.C. Ancell, E. Kashawlic, J.L. Schroeder, Evaluation of wind forecasts and observation impacts from variational and ensemble data assimilation for wind energy applications, Mon. Wea. Rev. 143 (2015) 3230–3245.

[8] J.L. Anderson, An adaptive covariance inflation error correction algorithm for ensemble filters, Tellus 59A (2007) 210–224.

[9] J.L. Anderson, S.L. Anderson, A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts, Mon. Wea. Rev. 127 (1999) 2741–2758.

[10] E. Andersson, H. Jarvinen, Variational quality control, Q. J. R. Meteor. Soc. 125 (1999) 697–722.

[11] A. Apte, C.K.R.T. Jones, The impact of nonlinearity in Lagrangian data assimilation, Nonlin. Proc. Geophys. 20 (2013) 329–341.

[12] A. Arakawa, V.R. Lamb, Computational design of the basic dynamical processes of the UCLA general circulation model, Methods Comput. Phys. 17 (1977) 173–265.

[13] A. Arakawa, S. Moorthi, Baroclinic instability in vertically discrete systems, J. Atmos. Sci. 45 (1988) 1688–1707.

[14] R. Arcucci, J. Zhu, S. Hu, Y.-K. Guo, Deep data assimilation: integrating deep learning with data assimilation, Appl. Sci. 11 (2021) 1114.

[15] N. Asadi, K.A. Scott, D.A. Clausi, Data fusion and data assimilation of ice thickness observations using a regularisation framework, Tellus A 71 (2019) 1564487.

[16] M.J. Atkins, The objective analysis of relative humidity, Tellus 26 (1974) 663–671.

[17] K.E. Atkinson, An Introduction to Numerical Analysis, John Wiley and Sons, New York, 1988.

[18] G. Backus, F. Gilbert, Uniqueness in the inversion of inaccurate gross Earth data, Philos. Trans. R. Soc. London Ser. A 266 (1970) 123–192.

[19] C. Bacour, P. Peylin, N. MacBean, P.J. Rayner, F. Delage, F. Chevallier, M. Weiss, J. Demarty, D. Santaren, F. Baret, D. Berveiller, E. Dufrêne, P. Prunet, Joint assimilation of eddy covariance flux measurement and FAPAR products over temperate forests within a process-orientated biosphere model, J. Geophys. Res. Biogeosci. 120 (2015) 1839–1857.

[20] J. Badger, B.J. Hoskins, Simple initial value problems and mechanisms for baroclinic growth, J. Atmos. Sci. 58 (2001) 39–49.

[21] N.L. Baker, R. Daley, Observation and background adjoint sensitivity in the adaptive observation-targeting problem, Q. J. R. Meteor. Soc. 126 (2000) 1431–1454.

[22] R.N. Bannister, A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics, Q. J. R. Meteor. Soc. 134 (2008) 1971–1996.

[23] E.H. Barker, Design of the navy's multivariate optimum interpolation analysis system, Wea. Forecast. 7 (1992) 220–231.
[24] S.L. Barnes, A technique for maximizing details in numerical weather map analysis, J. Appl. Meteor. 3 (1963) 396–409.
[25] S. Barnett, R.G. Cameron, Introduction to Mathematical Control Theory, Clarendon Press, Oxford, 1993.
[26] G.K. Batchelor, An Introduction to Fluid Dynamics, Cambridge University Press, Cambridge, 2000.
[27] J.R. Bates, A. McDonald, Multiply-upstream, semi-Lagrangian advective schemes: analysis and application to a multi-level primitive equation model, Mon. Wea. Rev. 110 (1982) 1831–1842.
[28] A. Beljadid, A. Mohammadin, M. Charron, C. Girard, Theoretical and numerical analysis of a class of semi-implicit semi-Lagrangian schemes potentially applicable to atmospheric models, Mon. Wea. Rev. 142 (2014) 4458–4476.
[29] A.F. Bennett, Inverse Modeling of the Ocean and Atmosphere, Cambridge University Press, Cambridge, 2002.
[30] K.H. Bergman, Multivariate analysis of temperature and winds using optimum interpolation, Mon. Wea. Rev. 107 (1979) 1423–1444.
[31] K.H. Bergman, T.N. Carlson, Objective analysis of aircraft data in tropical cyclones, Mon. Wea. Rev. 103 (1975) 431–444.
[32] P. Bergthórsson, B.R. Döös, Numerical weather map analysis, Tellus 7 (1955) 329–340.
[33] A. Bernigaud, S. Gratton, F. Lenti, E. Simon, L_p-norm regularization approaches in variational data assimilation, Q. J. R. Meteor. Soc. 147 (2021) 2067–2081.
[34] M. Bhargava, M. Danard, Normal mode initialization for simple models, Meteorol. Atmos. Phys. 60 (1996) 225–236.
[35] T. Bick, C. Simmer, S. Trömel, K. Wapler, H.-J. Hendricks Franssen, K. Stephan, U. Blahak, C. Schraff, H. Reich, Y. Zeng, R. Potthast, Assimilation of 3D radar reflectivities with an ensemble filter on the convective scale, Q. J. R. Meteor. Soc. 146 (2016) 1490–1504.
[36] G.J. Bierman, Factorization Methods for Discrete Sequential Estimation, Mathematics in Science and Engineering, vol. 128, Academic Press, New York, 1977.
[37] C.H. Bishop, The GIGG-EnKF Kalman filtering for highly skewed non-negative uncertainty distributions, Q. J. R. Meteor. Soc. 142 (2016) 1395–1412.
[38] C.H. Bishop, B.J. Etherton, S.J. Majumdar, Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects, Mon. Wea. Rev. 129 (2001) 420–436.
[39] C.H. Bishop, S. Frolov, D.R. Allen, D.D. Kuhl, K. Hoppel, The Local Ensemble Tangent Linear Model: an enabler for coupled model 4D-Var, Q. J. R. Meteor. Soc. 143 (2017) 1009–1020.
[40] C.H. Bishop, Z. Toth, Ensemble transformation and adaptive observations, J. Atmos. Sci. 56 (1999) 1748–1765.
[41] V. Bjerknes, Das problem der wettervorhersage, betrachtet vom stanpunkt der mechanik and der physik, Meteor. Z. 21 (1904) 1–7.
[42] M. Bocquet, P. Sakov, An iterative ensemble Kalman smoother, Q. J. R. Meteor. Soc. 140 (2014) 1521–1535.
[43] M. Bonavita, Exploring the structure of time correlated model errors in the ECMWF data assimilation system, Q. J. R. Meteor. Soc. 147 (2021) 3454–3471.
[44] M. Bonavita, E. Hólm, L. Isaken, M. Fisher, The evolution of the ECMWF hybrid data assimilation system, Q. J. R. Meteor. Soc. 142 (2016) 287–303.
[45] M. Bonavita, P. Lean, E. Holm, Nonlinear effects in 4D-Var, Nonlin. Proc. Geophys. 25 (2018) 713–729.
[46] S.-A. Boukabara, K.J. Garrett, W. Chen, F. Iturbide-Sánchez, C. Grassotti, C.E. Kongoli, R. Chen, Q. Liu, B. Yan, F. Weng, R.R. Ferraro, T.J. Kleespies, H. Meng, MiRS: an all-weather 1DVAR satellite data assimilation and retrieval system, IEEE Trans. Geosci. Remote Sens. 49 (2011) 3249–3272.
[47] N.E. Bowler, An assessment of GNSS radio occultation data produced by Spire, Q. J. R. Meteor. Soc. 146 (2020) 3772–3788.

[48] M. Buehner, Evaluation of a spatial/spectral covariance localization approach for atmospheric data assimilation, Mon. Wea. Rev. 131 (2012) 617–636.
[49] M. Buehner, R. McTaggart-Cowan, A. Beaulne, C. Charette, L. Garand, S. Heilliette, E. Lapalme, S. Laroche, S.R. Macpherson, J. Morneau, A. Zadra, Implementation of deterministic weather forecasting systems based on ensemble-variational data assimilation at Environment Canada. Part I: The global system, Mon. Wea. Rev. 143 (2015) 2532–2559.
[50] M. Buehner, A. Shlyaeva, Scale-dependent background-error covariance localisation, Tellus A 67 (2015) 28027.
[51] C. Buizza, C. Quilodrán Casas, P. Nadler, J. Mack, S. Marrone, Z. Titus, C. Le Cornec, E. Heylen, T. Dur, L. Baca Ruiz, C. Heaney, J. Amador Díz Lopez, K.S. Sech Kumar, R. Arcucci, Deep learning: integrating data assimilation and machine learning, J. Comp. Sci. 58 (2022) 101525.
[52] G. Burgers, P.J. Van Leeuwen, E. Evensen, Analysis schemes in the ensemble Kalman filter, Mon. Wea. Rev. 126 (1998) 1719–1724.
[53] G. Burgers, P.J. Van Leeuwen, G. Evensen, Analysis scheme in the Ensemble Kalman Filter, Mon. Wea. Rev. 126 (1998) 1719–1724.
[54] W.F. Campmell, C.H. Bishop, D. Hodyss, Vertical covariance localization for satellite radiances in ensemble Kalman filters, Mon. Wea. Rev. 138 (2010) 282–290.
[55] C. Cardinali, L. Isaksen, E. Andersson, Use and impact of automated aircraft data in a global 4DVAR data assimilation system, Mon. Wea. Rev. 131 (2003) 1865–1877.
[56] J.-F. Caron, M. Buehner, Scale-dependent background error covariance localization: evaluation in a global deterministic weather forecasting system, Mon. Wea. Rev. 146 (2018) 1376–1381.
[57] M.J. Carrier, H.E. Ngodock, P. Muscarella, S. Smith, Impact of assimilating surface velocity observation on the model sea surface height using the NCOM-4DVAR, Mon. Wea. Rev. 144 (2016) 1051–1068.
[58] F. Carse, M.J. Martin, A. Sellar, E.W. Blockley, Impact of assimilating temperature and salinity measurements from animal-borne sensors on FOAM ocean model fields, Q. J. R. Meteor. Soc. 141 (2015) 2934–2943.
[59] S.-F. Chang, Y.-C. Liou, J. Sun, S.-L. Tai, The implementation of the ice-phase microphysical process into four dimensional Variational Doppler Radar Analysis System (VDRAS) and its impact on parameter retrieval and quantitative precipitation nowcasting, J. Atmos. Sci. 73 (2016) 1015–1038.
[60] J.G. Charney, N.A. Phillips, Numerical integration of the quasi-geostrophic equations for barotropic and simple baroclinic, J. Meteor. 10 (1953) 71–99.
[61] A.J. Chorin, M. Morzfeld, X. Tu, Interpolation and iteration for nonlinear filters, Comm. Appl. Math. Comp. Sci. 5 (2010) 221–240.
[62] B.S. Chua, A.F. Bennett, An inverse ocean modeling system, Ocean Model. 3 (2001) 137–165.
[63] R.M. Clancy, P.A. Phoebus, K.D. Pollak, An operational global-scale ocean thermal analysis system, J. Atmos. Ocean. Tech. 7 (1990) 233–254.
[64] G.M. Clarke, D. Cooke, A Basic Course in Statistics, Oxford University Press, New York, 2004.
[65] R.T. Clarke, Extension of annual streamflow record by correlation with precipitation subject to heterogeneous errors, Wat. Res. Res. 15 (1979) 1081–1088.
[66] R.T. Clarke, Bivariate gamma distribution for extending annual stream flow records from precipitation, Wat. Res. Res. 16 (1980) 863–870.
[67] A.M. Clayton, A.C. Lorenc, D.M. Barker, Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the Met Office, Q. J. R. Meteor. Soc. 139 (2013) 1445–1461.
[68] S.E. Cohn, An introduction to estimation error theory, J. Meteor. Soc. Japan 75 (1997) 257–288.
[69] S.E. Cohn, A. Da Silva, J. Guo, M. Sienkiewicz, D. Lamich, Assessing the effects of data selection with the DAO physical-space statistical analysis system, Mon. Wea. Rev. 126 (1998) 2913–2926.
[70] S.E. Cohn, M. Sienkiewicz, R. Todling, A fixed-lag Kalman smoother for retrospective data assimilation, Mon. Wea. Rev. 122 (1994) 2838–2867.

[71] J.W. Cooley, O.W. Tukey, An algorithm for the machine calculation of complex Fourier Series, Math. Comput. 19 (1966) 297–301.
[72] G. Cortes, S. Girotto, M. Margulis, Snow process estimation over the extratropical Andes using a data assimilation framework integrating MERRA data and Landsat imagery, Wat. Res. Res. 52 (2016) 2582–2600.
[73] E. Cosme, J. Verron, P. Brasseur, J. Blum, D. Auroux, Smoothing problems in a Bayesian framework and their linear Gaussian solutions, Mon. Wea. Rev. 140 (2012) 683–695.
[74] R. Courant, D. Hilbert, Methods of Mathematical Physics, Volume 2: Partial Differential Equations, Wiley-VCH Verlag GmbH, Weinheim, Germany, 2008.
[75] P. Courtier, Dual formulation of four-dimensional variational assimilation, Q. J. R. Meteor. Soc. 123 (1997) 2449–2461.
[76] P. Courtier, O. Talagrand, Variational assimilation of meteorological observations with the direct and adjoint shallow-water equations, Tellus 42A (1990) 531–549.
[77] P. Courtier, J.-N. Thépaut, A. Hollingsworth, A strategy for operational implementation of 4D-VAR, using an incremental approach, Q. J. R. Meteor. Soc. 120 (1994) 1367–1387.
[78] N. Cressie, C.K. Wikle, Statistics for Spatio-temporal Data, Wiley, Hoboken, NJ, 2011.
[79] G.P. Cressman, An operational objective analysis system, Mon. Wea. Rev. 87 (1959) 367–374.
[80] E.L. Crow, K. Shimizu, Lognormal Distributions, Theory and Applications, Marcel Dekker, Inc., New York, 1988.
[81] L. Cucurull, J.C. Derber, R. Treadon, R. Purser, Assimilation of global positioning system radio occultation observations into NCEP's Global Data Assimilation system, Mon. Wea. Rev. 135 (2007) 3174–3193.
[82] R. Daley, Normal mode initialization, Rev. Geophys. Space Phys. 19 (1981) 450–468.
[83] R. Daley, The analysis of synoptic scale divergences by a statistical interpolation procedure, Mon. Wea. Rev. 113 (1985) 1066–1079.
[84] R. Daley, The lagged innovation covariance: a performance diagnostic for atmospheric data assimilation, Mon. Wea. Rev. 120 (1992) 178–196.
[85] R. Daley, Atmospheric Data Analysis, Cambridge University Press, Cambridge, UK, 1996.
[86] R. Daley, E. Barker, NAVDAS: formulation and diagnostics, Mon. Wea. Rev. 129 (2001) 869–883.
[87] I. Daužickaitė, A.S. Lawless, J.A. Scott, P.J. van Leeuwen, On time-parallel preconditioning for the state formulation of incremental weak constraint 4D-var, Q. J. R. Meteor. Soc. 147 (2021) 3521–3529.
[88] C. de Boor, B. Swatrz, Piecewise monotone interpolation, J. Approx. Theory 21 (1977) 411–416.
[89] D.P. Dee, Bias and data assimilation, Q. J. R. Meteor. Soc. 131 (2004) 3323–3343.
[90] D.P. Dee, L. Rukhovets, R. Todling, A.M. Da Silva, J.W. Larzon, An adaptive buddy check for observational quality control, Q. J. R. Meteor. Soc. 127 (2001) 2451–2471.
[91] D.P. Dee, S. Uppala, Variational bias correction of satellite radiance data in the ERA-Interim reanalysis, Q. J. R. Meteor. Soc. 135 (2004) 1830–1841.
[92] R. Delbourgo, J.A. Gregory, Shape preserving piecewise rational interpolation, SIAM J. Sci. Stat. Comput. 6 (1985) 967–976.
[93] J. Derber, F. Bouttier, A reformulation of the background error covariance in the ECMWF global data assimilation system, Tellus 51A (1999) 195–221.
[94] J.C. Derber, W. Wu, The use of TOVS cloud-cleared radiances in the NCEP SSI analysis system, Mon. Wea. Rev. 126 (1998) 2287–2299.
[95] G. Desroziers, L. Berre, B. Chapnik, P. Poli, Diagnosis of observation, background and analysis-errors statistics in observation space, Q. J. R. Meteor. Soc. 131 (2005) 3385–3396.
[96] G. Desroziers, J.-T. Camino, L. Berre, 4DEnVar: link wit 4D state formulation of variational assimilation and different possible implementations, Q. J. R. Meteor. Soc. 140 (2014) 2097–2110.
[97] J.-L. Devenon, Optimal control theory applied to an objective analysis of a tidal current mapping by HR radar, J. Atmos. Ocean. Tech. 7 (1990) 269–284.

[98] I.V. Djalalova, J. Olson, J.R. Carley, L. Bianco, J.M. Wilczak, Y. Pichugina, R. Banta, M. Marquis, J. Cline, The POWER experiment: impact of assimilation of a network of coastal wind profiling radars on simulating offshore winds in and above wind turbine layer, Wea. Forecast. 31 (2016) 1071–1091.
[99] D.I. Duncan, N. Bormann, E. Hólm, On the addition of microwave sounders and numerical weather prediction skill, Q. J. R. Meteor. Soc. 147 (2021) 3703–3718.
[100] E. Eady, Long waves and cyclone waves, Tellus 1 (1949) 33–52.
[101] A. Eddy, The objective analysis of atmospheric structure, J. Meteor. Soc. Japan 51 (1973) 450–457.
[102] A. Eliassen, The quasi-static equations of motion with pressure as independent variable, Geofys. Publikasjoner 17 (1949) 3.
[103] J.M. English, A.C. Kren, T.R. Peevey, Improving winter storm forecasts with Observing System Simulation Experiments (OSSEs). Part 2: Evaluating a satellite gap with idealized and targeted dropsondes, Earth Space Sci. 5 (2018) 176–196.
[104] E.S. Epstein, Stochastic dynamic prediction, Tellus 21 (1969) 739–759.
[105] R.M. Errico, N. Privè, W. Gu, Use of OSSE to evaluate background-error covariance estimated by the NMC method, Q. J. R. Meteor. Soc. 141 (2014) 611–618.
[106] R.M. Errico, K.D. Reader, An examination of the accuracy of the linearization of a mesoscale model with moist physics, Q. J. R. Meteor. Soc. 125 (1999) 169–195.
[107] R.N. Errico, R. Yang, M. Masutani, J. Wollen, Estimation of some characteristics of analysis error inferred from an observing system simulation experiment, Meteor. Z. 16 (2007) 695–708.
[108] R.N. Errico, R. Yang, N.C. Prive, K.-S. Tai, R. Todling, J. Guo, Development and validation of observing-system simulation experiments at NASA Global Modeling and Assimilation Office, Q. J. R. Meteor. Soc. 139 (2013) 1162–1178.
[109] G. Evensen, Using the extended Kalman filter with a multi-layer quasi-geostrophic ocean model, J. Geophys. Res. Oceans 97 (1992) 17905–17924.
[110] G. Evensen, Open boundary conditions for the extended Kalman filter with a quasi-geostrophic model, J. Geophys. Res. Oceans 98 (1993) 16529–16546.
[111] G. Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte-Carlo methods to forecast error statistics, J. Geophys. Res. Oceans 99 (C5) (1994) 10143–10162.
[112] G. Evensen, P.J. Van Leeuwen, Assimilation of Geosat altimeter data for the Agulhas current using the ensemble Kalman filter with a quasigeostrophic model, Mon. Wea. Rev. 124 (1996) 85–96.
[113] G. Evensen, P.J. van Leeuwen, An ensemble Kalman smoother for nonlinear dynamics, Mon. Wea. Rev. 128 (2000) 1852–1867.
[114] J.R. Eyre, Observation impact metrics in NWP: a theoretical study. Part I: Optimal systems, Q. J. R. Meteor. Soc. 147 (2021) 3180–3200.
[115] M. Fan, Y. Bali, L. Wang, L. Tang, L. Ding, Coupling the k-nearest neighbors and locally weighted linear regression with ensemble Kalman filter for data drive data assimilation, Open Geosci. 13 (2021) 1395–1413.
[116] Y.-C. Fang, T.J. Weingartner, R.A. Potter, P.R. Winsor, H. Statscewich, Quality assessment of HF radar-derived surface currents using optimal interpolation, J. Atmos. Ocean. Tech. 32 (2015) 282–296.
[117] A. Farchi, M. Bocquet, P. Laloyaux, M. Bonavita, Q. Malartic, A comparison of combined data assimilation and machine learning methods for offline and online model error corrections, J. Comp. Sci. 55 (2021) 101468.
[118] A. Farchi, P. Laloyaux, M. Bonavita, M. Bocquet, Using machine learning to correct model error in data assimilation and forecast applications, Q. J. R. Meteor. Soc. 147 (2021) 3067–3084.
[119] B. Farrell, The initial growth of disturbances in a baroclinic flow, J. Atmos. Sci. 39 (1982) 1663–1686.
[120] N. Feyeux, A. Vidard, M. Nodet, Optimal transport for variational data assimilation, Nonlin. Proc. Geophys. 25 (2018) 55–66.
[121] M. Fisher, Background error covariance modelling, in: ECMWF Seminar Series on Recent Developments in Data Assimilation for the Atmosphere and Ocean, 2003, pp. 45–64.

[122] M. Fisher, Estimation of entropy reduction and degrees of freedom for signal for large variational analysis systems, EVMWF Technical Memorandum 397, ECMWF, 2003.
[123] M. Fisher, Generalized frames on the sphere, with application to background error covariance modelling, in: ECMWF Seminar Series on Recent Developments in Numerical Methods for Atmospheric and Ocean Modelling, 2004, pp. 87–101.
[124] M. Fisher, S. Gürol, Parallelization in the time dimension of four-dimensional variational data assimilation, Q. J. R. Meteor. Soc. 142 (2017) 1136–1147.
[125] M. Fisher, D.J. Lary, Lagrangian four-dimensional variational data assimilation of chemical species, Q. J. R. Meteor. Soc. 121 (1995) 1681–1704.
[126] R. Fjørtoft, On a numerical method of integrating the barotropic vorticity equation, Tellus 4 (1952) 179–194.
[127] S.J. Fletcher, Numerical Approximations to Buoyancy Advection in the Eady Model, M.Sc. dissertation, University of Reading, Department of Mathematics, 1999.
[128] S.J. Fletcher, Higher Order Balance Conditions Using Hamiltonian Dynamics for Numerical Weather Prediction, Ph.D. thesis, University of Reading, Department of Mathematics, 2004.
[129] S.J. Fletcher, Mixed lognormal-Gaussian four-dimensional data assimilation, Tellus 62A (2010) 266–287.
[130] S.J. Fletcher, Data Assimilation for the Geosciences: From Theory to Applications, Elsevier, Amsterdam, Netherlands, 2017.
[131] S.J. Fletcher, Semi-Lagrangian Advection Methods and Their Application in Geoscience, Elsevier, Amsterdam, Netherlands, 2019.
[132] S.J. Fletcher, A.S. Jones, Multiplicative and additive incremental variational data assimilation for mixed lognormal-Gaussian errors, Mon. Wea. Rev. 142 (2014) 2521–2544.
[133] S.J. Fletcher, A.J. Kliewer, A.S. Jones, Quantification of optimal values for the parameters in lognormal variational data assimilation and their chaotic effects, Math. Geosci. 51 (2019) 187–207.
[134] S.J. Fletcher, G.E. Liston, C.A. Hiemstra, S.D. Miller, Assimilating MODIS and AMSR-E snow observations in a snow evolution model, J. Hydromet. 13 (2012) 1475–1492.
[135] S.J. Fletcher, M. Zupanski, A data assimilation method for log-normally distributed observational errors, Q. J. R. Meteor. Soc. 132 (2006) 2505–2519.
[136] S.J. Fletcher, M. Zupanski, A hybrid normal and lognormal distribution for data assimilation, Atmos. Sci. Lett. 7 (2006) 43–46.
[137] S.J. Fletcher, M. Zupanski, Implications and impacts of transforming lognormal variables into normal variables in VAR, Meteor. Z. 16 (2007) 755–765.
[138] S.J. Fletcher, M. Zupanski, A study of ensemble size and shallow water dynamics with the Maximum Likelihood Ensemble Filter, Tellus 60A (2008) 348–360.
[139] S.J. Fletcher, M. Zupanski, M.R. Goodliff, A.J. Kliewer, A.S. Jones, J.M. Forsythe, T.-C. Wu, M.J. Hossen, S. Van Loon, Lognormal and mixed Gaussian-lognormal Kalman filters, Mon. Wea. Rev. (2022), submitted for publication.
[140] J. Flowerdew, N.E. Bowler, Improving the use of observations to calibrate ensemble spread, Q. J. R. Meteor. Soc. 137 (2011) 467–482.
[141] J. Flowerdew, N.E. Bowler, On-line calibration of the vertical distribution of ensemble spread, Q. J. R. Meteor. Soc. 139 (2013) 1863–1874.
[142] J. Foster, M. Bevis, W. Raymond, Precipitable water and the lognormal distribution, J. Geophys. Res. Atmos. 111 (2006) D15102.
[143] A.M. Fowler, A.S. Lawless, An idealized study of coupled atmosphere-ocean 4D-Var in the presence of model error, Mon. Wea. Rev. 144 (10) (2016) 4007–4030.
[144] M.A. Freitag, N.K. Nichols, C.J. Budd, L_1-regularisation for ill-posed problems in variational data assimilation, Proc. Appl. Math. Mech. 10 (2010) 665–668.
[145] M.A. Freitag, N.K. Nichols, C.J. Budd, Resolution of sharp fronts in the presence of model error in variational data assimilation, Q. J. R. Meteor. Soc. 139 (2013) 742–757.

[146] F.N. Fritsch, R.E. Carlson, Monotone piecewise cubic interpolation, SIAM J. Numer. Anal. 2 (1980) 238–246.
[147] S. Frolov, C.H. Bishop, T. Holt, J. Cummings, D. Kuhl, Facilitating strongly coupled ocean-atmosphere data assimilation with an interface solver, Mon. Wea. Rev. 144 (2016) 3–20.
[148] F. Gaillard, T. Reynaud, V. Thierry, N. Kolodziejczyk, K. Von Schuckmann, In Situ-Based reanalysis of the global ocean temperature and salinity with ISAS: variability of the heat content and steric height, J. Clim. 29 (2016) 1305–1323.
[149] L.S. Gandin, Objective analysis of meteorological fields, translated from Russian by the Israeli Program for Scientific Translations, 1965.
[150] P.R. Garabedian, Partial Differential Equations, American Mathematical Society, Providence, RI, 1998.
[151] L.C. Gardner, R.W. Schunk, L. Scherliess, J.J. Sojka, L. Zhu, Global Assimilation of Ionospheric Measurements-Global Markov model: improved specifications with multiple data types, Space Weather 12 (2014) 675–688.
[152] G. Gaspari, S. Cohn, Construction of correlation functions in two and three dimensions, Q. J. R. Meteor. Soc. 125 (1999) 723–757.
[153] P. Gauthier, C. Charette, L. Fillion, P. Koclas, S. Laroche, Implementation of a 3D variational data assimilation system at the Canadian Meteorological Centre. Part I: The global analysis, Ocean Atmos. 37 (1999) 103–156.
[154] P. Gauthier, M. Tanguay, S. Laroche, S. Pellering, J. Morneau, Extension of a 3D-Var to 4D-Var: implementation of 4D-Var at the Meteorological Service of Canada, Mon. Wea. Rev. 135 (2007) 2339–2354.
[155] A. Gelb, Applied Optimal Estimation, MIT Press, Boston, 1974.
[156] M. Girotto, G.J.M. De Lannoy, R.H. Reichle, M. Rodell, Assimilation of gridded terrestrial water storage observations from GRACE into a land surface model, Wat. Res. Res. 52 (2016) 4164–4183.
[157] G.H. Golub, C.F. Van Loan, Matrix Computations, third ed., The John Hopkins University Press, Baltimore, MD, USA, 1996.
[158] M. Goodliff, S. Fletcher, A. Kliewer, J. Forsythe, A. Jones, Detection of non-Gaussian behavior using machine learning techniques: a case study on the Lorenz 63 model, J. Geophys. Res. Atmos. 125 (2020) e2019JD031551.
[159] M.R. Goodliff, S.J. Fletcher, A.J. Kliewer, A.S. Jones, J.M. Forsythe, Non-Gaussian detection using machine learning with data assimilation applications, Earth Space Sci. 9 (2022) e2021EA001908.
[160] S. Gravel, A. Staniforth, J. Côté, A stability analysis of a family of baroclinic semi-Lagrangian forecast models, Mon. Wea. Rev. 117 (1989) 130–137.
[161] A. Griffith, N.K. Nichols, Adjoint methods for treating model error in data assimilation, in: Numerical Methods for Fluid Dynamics, Institute for CFD, Oxford, 1998, pp. 335–344.
[162] A.R. Gusman, A.F. Sheehan, K. Satake, M. Heidarzadeh, I.E. Mulia, T. Maeda, Tsunami data assimilation of Cascadia seafloor pressure gauge records from the 2012 Haida Gwaii earthquake, Geophys. Res. Lett. 43 (2016) 4189–4196.
[163] G.R. Halliwell Jr., M. Mehrai, L.K. Shay, V.H. Kourafalou, H. Kang, H.-S. Kimm, J. Dong, R. Atlas, OSSE quantitative assessment of rapid-response prestorm ocean survey to improve coupled tropical cyclone prediction, J. Geophys. Res. Oceans 122 (2017) 5729–5748.
[164] T.M. Hamil, C. Snyder, A hybrid ensemble Kalman filter—3D Variational analysis scheme, Mon. Wea. Rev. 128 (2000) 2905–2919.
[165] T.M. Hamil, J.S. Whitaker, C. Snyder, Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter, Mon. Wea. Rev. 129 (2001) 2776–2790.
[166] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, 2nd edition, Springer, New York, NY, 2008.
[167] W. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrica 57 (1970) 97–109.

[168] C.M. Hayden, R.J. Purser, Recursive filter objective analysis of meteorological fields: applications to NESDIS operational processing, J. Appl. Meteor. 34 (1995) 3–15.
[169] G.P. Hayden, Experiments in the four-dimensional assimilation of Nimbus 4 SIRS data, J. Appl. Meteor. 12 (1973) 425–436.
[170] S. Haykin, Kalman Filtering and Neural Networks, John Wiley and Sons, New York, 2001.
[171] R. Heikes, D.A. Randall, Numerical integration of the shallow water equations on a twisted icosahedral grid: Part I: Basic design and results of tests, Mon. Wea. Rev. 123 (1995) 1862–1880.
[172] R. Heikes, D.A. Randall, Numerical integration of the shallow water equations on a twisted icosahedral grid: Part II: A detailed description of the grid and an analysis of numerical accuracy, Mon. Wea. Rev. 123 (1995) 1881–1887.
[173] C.C. Heyde, On a property of the lognormal distribution, J. R. Stat. Soc. Ser. B. 25 (1963) 392–393.
[174] D. Hodyss, Ensemble state estimation for nonlinear systems using polynomial expansions in the innovation, Mon. Wea. Rev. 139 (2011) 3571–3588.
[175] D. Hodyss, Accounting for skewness in ensemble data assimilation, Mon. Wea. Rev. 140 (2012) 2346–2358.
[176] D. Hodyss, N.K. Nichols, The error of representation: basic understanding, Tellus 67 (2015) 24822.
[177] J.J. Hoelzemann, H. Elbern, A. Ebel, PSAS and 4D-var data assimilation for chemical state analysis by urban and rural observation sites, Phys. Chem. Earth 10 (2001) 807–812.
[178] J. Hoke, R. Anthes, The initialization of numerical models by a dynamic relaxation technique, Mon. Wea. Rev. 104 (1976) 1551–1556.
[179] A. Hollingsworth, P. Lönnberg, The statistical structure of short-range forecast errors as determined from radiosonde data. Part I: The wind field, Tellus 38A (1986) 111–136.
[180] B.J. Hoskins, The geostrophic momentum approximation and the semigeostrophic equations, J. Atmos. Sci. 32 (1975) 233–242.
[181] D. Hotta, T.-C. Chen, E. Kalnay, Y. Ota, T. Miyoshi, Proactive QC: a fully flow-dependent quality control scheme based upon EFSO, Mon. Wea. Rev. 145 (2017) 3331–3354.
[182] D.D. Houghton, Derivation of the elliptic condition for the balance equation in spherical coordinates, J. Atmos. Sci. 25 (1968) 927–928.
[183] P.L. Houtekamer, H.L. Mitchell, Data assimilation using an ensemble Kalman filter technique, Mon. Wea. Rev. 126 (1998) 796–811.
[184] P.L. Houtekamer, H.L. Mitchell, A sequential ensemble Kalman filter for atmospheric data assimilation, Mon. Wea. Rev. 129 (2001) 123–137.
[185] J. Huang, L. Tian, S. Liang, I. Becker-Reshef, Y. Huang, W. Su, J. Fan, W. Wu, Improving winter wheat yield estimation by assimilation of leaf area index from Landsat and Modis data into the WOFOST model, Agric. For. Meteorol. 204 (2015) 106–121.
[186] X.-Y. Huang, P. Lynch, Diabatic digital filtering initialization: application to the HIRLAM, Mon. Wea. Rev. 121 (1993) 2719–2726.
[187] P.J. Huber, Robust regression: asymptotics, conjectures, and Monte Carlo, Ann. Stat. 1 (1973) 799–821.
[188] B.R. Hunt, E.J. Kostelich, I. Szunyogh, Efficient data assimilation for spatiotemporal chaos, Physica D 230 (2007) 112–126.
[189] J.M. Hyman, Accurate monotonicity preserving cubic interpolation, SIAM J. Sci. Stat. Comput. 4 (1983) 645–654.
[190] K. Ide, P. Courtier, M. Ghil, A.C. Lorenc, Unified notation for data assimilation: operational, sequential and variational, J. Meteor. Soc. Japan 75 (1997) 181–189.
[191] K. Ide, L. Kuznetsoz, C.K.R.T. Jones, Lagrangian assimilation for point vortex systems, J. Turbul. 3 (2002), https://doi.org/10.1088/1468-5248/3/1/053.
[192] I. Iermano, A.M. Moore, Z. Zamianchi, Impact of a 4-dimensional variational data assimilation ocean model of southern Tyrrhenian Sea, J. Mar. Syst. 154 (2015) 2934–2943.
[193] N.B. Ingleby, A.C. Lorenc, Bayesian quality control using multivariate normal distributions, Q. J. R. Meteor. Soc. 119 (1993) 1195–1225.

[194] L. Isaksen, M. Bonavita, R. Buizza, M. Fisher, J. Haseler, M. Leutbecher, L. Raynaud, Ensemble of Data assimilations at ECMWF, Technical Memorandum 636, ECMWF, 2010.
[195] K. Ito, K. Xiong, Gaussian filters for nonlinear filtering problems, IEEE Trans. Autom. Control 45 (2000) 910–927.
[196] D.D. Jackson, The use of a priori data to resolve non-uniqueness in linear inversion, Geophys. J. R. Astron. Soc. 57 (1979) 137–157.
[197] S. Jahanbakhshi, M.R. Pishvaie, R.B. Boozarjomehry, Joint estimation of absolute and relative permeabilities using ensemble-based Kalman filter, J. Nat. Gas Sci. Eng. 26 (2015) 1232–1245.
[198] A.H. Jazwinski, Stochastic Processes and Filtering Theory, Academic Press, New York, 1970.
[199] H. Jin, A. Li, J. Wang, Y. Bo, Improving of spatially and temporally continuous crop leaf area index by integration of CRES-Maize model and MODIS data, Eur. J. Argon. 78 (2016) 1–12.
[200] S. Julier, The scaled unscented transformation, in: Proc. 2002 American Control Conf., Vol. 6, Anchorage, AK, IEEE, 2002, pp. 4555–4559.
[201] S. Julier, J. Uhlmann, H. Durrant-Whyte, A new approach for filtering nonlinear systems, in: Proceeding 1995 American Control Conference, Seattle, WA, IEEE, 1995, pp. 1628–1632.
[202] N.-Y. Jung, S. Kim, Y. Jo, Representer-based variational data assimilation in a spectral element shallow water model on the cubed-sphere grid, Tellus 66A (2014) 24493.
[203] A. Kageyama, T. Sato, Yin-Yang grid: an overset grid in spherical geometry, Geochem. Geophys. Geosyst. 5 (2004) 1–15.
[204] K. Kageyama, Dissection of a sphere and Yin-Yang grids, J. Earth Sim. 3 (2005) 20–28.
[205] R.E. Kalman, A new approach to linear filtering and prediction problems, Trans. ASME J. Basic Eng. 82 (1960) 35–45.
[206] R.E. Kalman, R.S. Bucy, New results in linear filtering and prediction theory, AMSRE J. Basic Eng. (1961) 95–108.
[207] R.E. Kalman, R.S. Bucy, New results in linear filtering and prediction, Trans. ASME J. Basic Eng. 83 (1961) 95–107.
[208] E. Kalnay, Atmospheric Modeling, Data Assimilation and Predictability, Cambridge University Press, Cambridge, UK, 2003.
[209] H. Kato, A. Yoshizawa, G. Ueno, S. Obayashi, A data assimilation methodology for reconstructing turbulent flows around aircraft, J. Comput. Phys. 283 (2015) 559–581.
[210] J.D. Kepert, Covariance localisation and balance in an ensemble Kalman filter, Q. J. R. Meteor. Soc. 135 (2009) 1157–1176.
[211] S.Q. Kidder, T.H. Vonder Haar, Satellite Meteorology: An Introduction, Academic Press, San Diego, 1995.
[212] S. Kim, G.L. Eyink, J.M. Resrepo, F.J. Alexander, G. Johnson, Ensemble filtering for nonlinear dynamics, Mon. Wea. Rev. 131 (2003) 2586–2594.
[213] S.-M. Kim, H.M. Kim, Forecast sensitivity observation impact in the 4DVAR and Hybrid-4DVAR data assimilation system, J. Atmos. Ocean. Tech. 36 (2019) 1563–1575.
[214] R. Kimura, Numerical weather prediction, J. Wind Eng. Ind. Aerodyn. 90 (2002) 1403–1414.
[215] R.R. King, D.J. Lea, M.J. Martin, I. Mirouze, J. Heming, The impact of Argo observations in a global weakly coupled ocean-atmosphere data assimilation and short-range prediction system, Q. J. R. Meteor. Soc. 146 (2020) 401–414.
[216] R.E. Kistler, R.D. McPherson, On the use of a local wind correction technique in four-dimensional data assimilation, Mon. Wea. Rev. 103 (1975) 445–449.
[217] D.T. Kleist, K. Ide, An OSSE-based evaluation of hybrid variational-ensemble data assimilation for the NCEP GFS. Part I: System description and 3D-hybrid results, Mon. Wea. Rev. 143 (2015) 433–451.
[218] D.T. Kleist, K. Ide, An OSSE-based evaluation of hybrid variational-ensemble data assimilation for the NCEP GFS. Part II: 4DEnVar and hybrid variants, Mon. Wea. Rev. 143 (2015) 452–470.
[219] D.T. Kleist, D.F. Parrish, J.C. Derber, T. Treadon, W. Wu, S. Lord, Introduction of the GSI into NCEP global data assimilation system, Wea. Forecast. 24 (2009) 1691–1705.

[220] A.J. Kliewer, S.J. Fletcher, A.S. Jones, J.M. Forsthye, Identifying non-normal and lognormal characteristics of temperature, mixing ratio, surface pressure, and wind for data assimilation systems, Nonlin. Proc. Geophys. Discussions 2 (2015) 1363–1405.
[221] A.J. Kliewer, S.J. Fletcher, A.S. Jones, J.M. Forsthye, Comparison of Gaussian, logarithmic transform and mixed distribution Gaussian-log-normal distribution based 1DVAR microwave temperature-water vapour mixing ration retrievals, Q. J. R. Meteor. Soc. 142 (2016) 274–286.
[222] S. Kobayashi, Y. Ota, Y. Harada, A. Ebita, M. Moriya, H. Onoda, K. Onogi, H. Kamahori, C. Kobayashi, H. Endo, K. Miyaoka, K. Takahashi, The JRA-55 reanalysis: general specifications and basic characteristics, J. Meteor. Soc. Japan 93 (2015) 5–48.
[223] D. Kondrashov, M. Ghil, Y. Shprits, Lognormal Kalman filter for assimilating phase space density data in the radiation belts, Space Weather 9 (2011) S11006.
[224] S. Kotz, N. Balakrishnan, N.L. Johnson, Continuous Multivariate Distributions, Volume 1: Models and Applications, Wiley, New York, 2000.
[225] M. Krzywinski, N. Altman, Classification and regression trees, Nat. Methods 14 (2017) 757–7568.
[226] D.D. Kuhl, T.E. Rosmond, C.H. Bishop, J. McLay, N.L. Baker, Comparison of hybrid ensemble/4DVar and 4DVar within the NAVDAS-AR data assimilation framework, Mon. Wea. Rev. 141 (2013) 2740–2758.
[227] S. Kumar Das, A.J. Weaver, Semi-Lagrangian advection algorithms for ocean circulation models, J. Atmos. Ocean. Tech. 12 (1995) 935–950.
[228] S.V. Kumar, B.F. Zaitchik, C.D. Peters-Lidard, M. Rodell, R. Reichle, B. Li, M. Jasinski, D. Mocko, A. Getirana, G. De Lannoy, M.H. Cosh, C.R. Hain, M. Anderson, K.R. Arsenault, Y. Xia, M. Ek, Assimilation of gridded GRACE terrestrial water storage estimates in the North American Land Data Assimilation System, J. Hydromet. 17 (2016) 1951–1972.
[229] Y. Kurihara, Numerical integration of the primitive equations on primitive grids, Mon. Wea. Rev. 93 (1965) 399–415.
[230] Y. Kurihara, On the use of implicit and iterative methods for the time integration of the wave equation, Mon. Wea. Rev. 93 (1965) 33–46.
[231] L. Kuznetsov, K. Ide, C.K.R.T. Jones, A method for assimilation of Lagrangian data, Mon. Wea. Rev. 131 (2003) 2247–2260.
[232] T. Lagarde, Nouvelle approache des methodes d'assimialtion de données les algorithms de point selle, Ph.D. thesis, Université Paul Sabatier, Toulouse, 2000.
[233] S. Lakshmivarahan, J.M. Lewis, D. Phan, Data assimilation as a problem in optimal tracking: application of Pontryagin's minimum principle to atmospheric science, J. Atmos. Sci. 70 (2013) 1257–1277.
[234] P. Laloyaux, M. Balmaseda, D. Dee, K. Mogensen, P. Janssen, A coupled data assimilation system for climate reanalysis, Q. J. R. Meteor. Soc. 142 (2016) 65–78.
[235] P. Laloyaux, M. Bonavita, M. Dahoui, J. Farnan, S. Healy, E. Hólm, S.T.K. Lang, Towards an unbiased stratospheric analysis, Q. J. R. Meteor. Soc. 146 (2020) 2392–2409.
[236] P. Laloyaux, J.-N. Thèpaut, D. Dee, Impact of scatterometer surface wind data in the ECMWF coupled assimilation system, Mon. Wea. Rev. 144 (2016) 1203–1217.
[237] R.H. Langland, N.L. Baker, Estimation of observation impact using the NRL atmospheric variational data assimilation adjoint system, Tellus 56A (2004) 189–201.
[238] T. Lauvaux, N.L. Miles, A. Deng, S.J. Richardson, M.O. Cambaliza, K.J. Davis, B. Gaudet, K.R. Gurney, J. Huang, D. O'Keefe, Y. Song, A. Karion, T. Oda, R. Patarasuk, I. Razlivanov, D. Sarmiento, P. Shepson, C. Sweeney, J. Turnbull, K. Wu, High-resolution atmospheric inversion of CO_2 emissions during the dormant season of the Indianapolis Flux Experiment (INFLUX), J. Geophys. Res. Atmos. 121 (2016) 5213–5236.
[239] A. Lavrov, Stability and dispersion analysis of semi-Lagrangian methods with Hermite interpolation, Numer. Heat Transf. Part B 155 (3) (2009) 177–195.
[240] A.S. Lawless, A perturbation forecast model and its adjoint, in: Preprints of the 11th Conference on Numerical Weather Prediction, American Meteorological Society, 1996.

[241] A.S. Lawless, Development of linear models for data assimilation in numerical weather prediction, Ph.D. thesis, Department of Mathematics, University of Reading, 2001.
[242] A.S. Lawless, A note on the analysis error associated with 3D-FGAT, Q. J. R. Meteor. Soc. 136 (2010) 1094–1098.
[243] A.S. Lawless, N.K. Nichols, S.P. Ballard, A comparison of two methods for developing the linearization of a shallow-water model, Q. J. R. Meteor. Soc. 129 (2003) 1237–1254.
[244] A.S. Lawless, N.K. Nichols, C. Boess, A. Bunse-Gerstner, Using model reduction methods within incremental 4D-Var, Mon. Wea. Rev. 136 (2008) 1511–1522.
[245] F.-X. Le Dimet, O. Talagrand, Variational algorithm for analysis and assimilation adjustment problem with advective constraints, Tellus 38A (1986) 97–110.
[246] D.J. Lea, I. Mirouze, M.J. Martin, R.R. King, A. Hines, D. Walters, M. Thurlow, Assessing a new coupled data assimilation system based on the Met Office coupled atmosphere-land-ocean-sea ice model, Mon. Wea. Rev. 143 (2015) 4678–4694.
[247] K. Lee, S. Jung, J. Choe, Ensemble smoother with clustered covariance for 3D channelized reservoir with geological uncertainty, J. Petrol. Sci. Eng. 145 (2016) 423–435.
[248] T. Lefebvre, H. Bruyninckx, J. De Schutter, Comment on "A new method for the nonlinear transformation of means and covariances in filters and estimators, IEEE Trans. Autom. Control (2002) 1406–1409.
[249] S. Legler, T. Janjić, Combining data assimilation and machine learning to estimate parameters of a convective-scale model, Q. J. R. Meteor. Soc. 148 (2022) 860–874.
[250] J. Levin, H.G. Arango, B. Laughlin, J. Wilkin, A.M. Moore, The impact of remote sensing observations on cross-shelf transport estimates from 4D-Var analyses of the Mid-Atlantic Bight, Adv. Space Res. 68 (2021) 553–570.
[251] J.M. Lewis, J.C. Derber, The use of adjoints equations to solve a variational adjustment problem with advective constraints, Tellus 37A (1985) 309–322.
[252] J.M. Lewis, S. Lakshmivarahan, Sasaki's pivotal contribution: calculus of variation applied to weather map analysis, Mon. Wea. Rev. 136 (2008) 3553–3567.
[253] Y. Li, I.M. Navon, P. Courtier, P. Gauthier, Variational data assimilation with a semi-Lagrangian semi-implicit global shallow-water equation model and its adjoint, Mon. Wea. Rev. 121 (1993) 1759–1769.
[254] X. Liang, X. Zheng, S. Zhang, G. Wu, Y. Dai, Y. Li, Maximum likelihood estimation of inflation factors on error covariance matrices for ensemble Kalman filter assimilation, Q. J. R. Meteor. Soc. 138 (2012) 263–273.
[255] D. Liu, A.K. Mishra, Z. Yu, H. Lü, Y. Li, Support vector machine and data assimilation framework for groundwater level forecasting using GRACE satellite data, J. Hydrol. 603 (2021) 126929.
[256] P. Lönnberg, A. Hollingsworth, The statistical structure of short-range forecast errors as determined from radiosonde data. Part II: The covariance of height and wind errors, Tellus 38A (1986) 137–161.
[257] A. Lorenc, Recommended nomenclature for EnVar data assimilation methods, in: Research Activities in Atmospheric and Oceanic Modeling, WGNE/WMO, 2013.
[258] A.C. Lorenc, A global three-dimensional multivariate statistical interpolation scheme, Mon. Wea. Rev. 109 (1981) 701–721.
[259] A.C. Lorenc, Analysis methods for numerical weather prediction, Q. J. R. Meteor. Soc. 112 (1986) 1177–1194.
[260] A.C. Lorenc, Optimal nonlinear objective analysis, Q. J. R. Meteor. Soc. 114 (1988) 205–240.
[261] A.C. Lorenc, Modelling of error covariances by 4D-Var data assimilation, Q. J. R. Meteor. Soc. 129 (2003) 3167–3182.
[262] A.C. Lorenc, S.P. Ballard, R.S. Bell, N.B. Ingleby, P.L.F. Andrews, D.M. Barker, J.R. Bray, A.M. Clayton, T. Dalby, D. Li, T.J. Payne, F.W. Saunders, The Met. Office global three dimensional variational data assimilation scheme, Q. J. R. Meteor. Soc. 126 (2000) 2991–3012.
[263] A.C. Lorenc, R.S. Bell, B. Macpherson, The Meteorological Office analysis correction data assimilation scheme, Q. J. R. Meteor. Soc. 117 (1991) 59–89.

[264] A.C. Lorenc, N.E. Bowler, A.M. Clayton, S.R. Pring, D. Fairbairn, Comparison of hybrid-4DEnVar and hybrid-4DVar data assimilation methods for global NWP, Mon. Wea. Rev. 143 (2015) 212–229.
[265] A.C. Lorenc, O. Hammon, Objective quality control of observations using Bayesian methods: theory, and a practical implementation, Q. J. R. Meteor. Soc. 114 (1988) 515–543.
[266] A.C. Lorenc, M. Jardak, A comparison of hybrid variational data assimilation methods for global NWP, Q. J. R. Meteor. Soc. 144 (2018) 2478–2760.
[267] A.C. Lorenc, R. Marriott, Forecast sensitivity observation impact in the Met Office global numerical weather prediction system, Q. J. R. Meteor. Soc. 140 (2014) 209–224.
[268] A.C. Lorenc, F. Rawlins, Why does 4D-Var beat 3D-Var?, Q. J. R. Meteor. Soc. 131 (2005) 3247–3257.
[269] E.N. Lorenz, Energy and numerical weather prediction, Tellus 12A (1960) 364–373.
[270] E.N. Lorenz, Deterministic nonperiodic flow, J. Atmos. Sci. 20 (1963) 130–141.
[271] E.N. Lorenz, Predictability: a problem partly solved, in: ECMWF Proceedings of Seminar on Predictability, Reading, UK, 1996, pp. 1–18.
[272] S. Lu, H.X. Lin, A.W. Heemink, G. Fu, A.J. Segers, Estimation of volcanic ash emissions using trajectory-based 4D-Var data assimilation, Mon. Wea. Rev. 144 (2016) 575–589.
[273] P. Lynch, X.-Y. Huang, Initialization of the HIRLAM model using a digital filter, Mon. Wea. Rev. 120 (1992) 1019–1034.
[274] B. Machenhauer, On the dynamics of gravity oscillations in a shallow water equation model, with application to normal mode initialization, Contrib. Atmos. Phys. 50 (1977) 253–271.
[275] J. Magnusson, D. Gustafsson, F. Hüsler, T. Jonas, Assimilation of point SWE data into a distributed snow cover model comparing two contrasting methods, Wat. Res. Res. 50 (2014) 7816–7835.
[276] S.E. Margulis, G. Cortês, M. Girotto, M. Durand, A Landsat-Era Sierra Nevada snow reanalysis (1985–2015), J. Hydromet. 17 (2016) 1203–1221.
[277] L. Marshall, D. Nott, A. Shara, Hydrological model selection: a Bayesian alternative, Wat. Res. Res. 41 (2005) W10422.
[278] J.P. Mattern, M. Dowd, K. Fennel, Particle filter based data assimilation for a three dimensional biological ocean model and satellite observations, J. Geophys. Res. 119 (2013) 2749–2760.
[279] A. McDonald, Accuracy of multiply-upstream, semi-Lagrangian advective schemes, Mon. Wea. Rev. 112 (1984) 1267–1275.
[280] A. McDonald, A semi-Lagrangian and semi-implicit two time-level integration scheme, Mon. Wea. Rev. 114 (1986) 824–830.
[281] A. McDonald, Semi-Lagrangian methods, in: ECMWF Seminar Series on Numerical Methods in Atmospheric Modelling, 1991, pp. 257–277.
[282] A. McDonald, An examination of alternative extrapolation to find the departure point position in a two-time-level semi-Lagrangian integration, Mon. Wea. Rev. 127 (1999) 1985–1993.
[283] A. McDonald, J.R. Bates, Improving the estimate of the departure point position in a two-time level semi-Lagrangian and semi-implicit scheme, Mon. Wea. Rev. 115 (1987) 737–739.
[284] A. McDonald, J.R. Bates, Semi-Lagrangian integration of a gridpoint shallow water model on the sphere, Mon. Wea. Rev. 121 (1993) 815–824.
[285] A. McDonald, J. Haugen, A two-time-level, three-dimensional semi-Lagrangian, semi-implicit limited-area gridpoint model of the primitive equations, Mon. Wea. Rev. 120 (1992) 2603–2621.
[286] L.A. McGee, S.F. Schmidt, Discovery of the Kalman filter as a practical tool for aerospace and industry, NASA Technical Memorandum 86847, NASA, 1985.
[287] M.E. McIntyre, I. Roulstone, Hamiltonian balanced models: constrained, slow manifolds and velocity-splitting, Forecasting Research Scientific Paper 41, Met. Office, 1996.
[288] M.E. McIntyre, I. Roulstone, Are there higher-accuracy analogues of semi-geostrophic theory?, in: J. Norbury, I. Roulstone (Eds.), Large-Scale Atmospheric-Ocean Dynamics, II: Geometric Methods and Models, Cambridge University Press, Cambridge, 2002, pp. 301–364.

[289] J.G. McLay, C.H. Bishop, C.A. Reynolds, Evaluation of the ensemble transform analysis perturbation scheme at NRL, Mon. Wea. Rev. 136 (2008) 1093–1108.
[290] J.G. McLay, C.H. Bishop, C.A. Reynolds, A local formulation of the Ensemble Transform (ET) analysis perturbation scheme, Wea. Forecast. 25 (2010) 985–993.
[291] T. McNally, M. Bonavita, J.-N. Thépaut, The role of satellite data in the forecasting of hurricane Sandy, Mon. Wea. Rev. 142 (2014) 634–646.
[292] O. Melnichenko, P. Hacker, N. Maximenko, G. Lagerloef, J. Potemra, Optimum interpolation analysis of Aquarius sea surface salinity, J. Geophys. Res. Oceans 121 (2016) 602–616.
[293] R. Mènard, R. Daley, The application of Kalman smoother theory to the estimation of 4DVAR error statistics, Tellus 48A (1996) 221–237.
[294] V.G. Merkin, D. Kondrashov, M. Ghil, B.J. Anderson, Data assimilation of low-altitude magnetic perturbations into a global magnetosphere model, Space Weather 14 (2016) 165–184.
[295] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machine, J. Chem. Phys. 21 (1953) 1087–1091.
[296] F. Miao, L. Xin, Paleoclimate data assimilation: its motivation, progress and prospects, Sci. China (2016) 1–4.
[297] D. Mignac, M.J. Martin, E. Fiedler, E.W. Blockley, N. Fournier, Improving the Met OFfice's Forecast Ocean Assimilation Model (FOAM) with the assimilation of satellite-derived sea-ice thickness data from CryoSat-2 and SMOS in the Arctic, Q. J. R. Meteor. Soc. 148 (2022) 2934–2943.
[298] T. Milewski, M.S. Bourqui, Potential of an ensemble Kalman smoother for stratospheric chemical-dynamical data assimilation, Tellus 65A (2013) 18541.
[299] S.D. Miller, C.L. Combs, S.Q. Kidder, Assessing moonlight availability for nighttime environmental applications by low-light visible polar-orbiting satellite sensors, J. Atmos. Ocean. Tech. 29 (2012) 538–557.
[300] S.D. Miller, A.K. Heidinger, M. Sengupta, Physically based satellite methods, in: J. Kleissl (Ed.), Solar Energy Forecasting and Resource Assessment, Academic Press, New York, USA, 2013, pp. 49–79.
[301] S.D. Miller, S.P. Mills, C.D. Elvidge, D.T. Lindsey, T.F. Lee, Suomi satellite brings to light a unique frontier of nighttime environmental sensing capabilities, Proc. Natl. Acad. Sci. U. S. A. 109 (2012) 15706–15711.
[302] L. Mingari, A. Folch, A.T. Prata, F. Pardini, G. Macedonio, A. Costa, Data assimilation of volcanic aerosol observations using FALL3D+PDAF, Atmos. Chem. Phys. 22 (2022) 1773–1792.
[303] H.L. Mitchell, P.L. Houtekamer, G. Pellerin, Ensemble size, balance and model-error representation in an ensemble Kalman filter, Mon. Wea. Rev. 130 (2002) 2791–2808.
[304] G. Monge, Mémoire sur la théorie des déblais et des remblais, Histoire de l'Académie Royale des Sciences de Paris, avec les Mémoires de Mathématique et de Physique pour la même année, 1781, pp. 666–704.
[305] P. Morel, G. Lefevre, G. Rabreau, On initialization and non-synoptic data assimilation, Tellus 23 (1971) 197–205.
[306] H. Morrison, M. van Lier-Walqui, A.M. Fridlind, W.W. Grabowski, J.Y. Harrington, C. Hoose, A. Korolev, M.R. Kumjian, J.A. Milbrandt, H. Pawloska, D.J. Posselt, O.P. Prat, K.J. Reimel, S.-I. Shima, B. van Diedenhoven, L. Xue, Confronting the challenge of modeling cloud and precipitation microphysics, Mon. Wea. Rev. 87 (2020) e2019MS001689.
[307] M. Morzfeld, S. Hodyss, J. Poterjoy, Variational particle smoothers and their localization, Q. J. R. Meteor. Soc. 144 (2018) 806–825.
[308] M. Morzfeld, T. Xu, E. Atkins, A.J. Chorin, A random map implementation of implicit filters, J. Comput. Phys. 231 (2012) 2049–2066.
[309] K. Mosegarad, A. Tarantola, Monte Carlo sampling of solutions to inverse problems, J. Geophys. Res. 100 (1995) B03097.
[310] B.G. Mullett, J. Korenaga, S.-I. Karato, Marko chain Monte Carlo inversion for the rheology of olivine single crystals, J. Geophys. Res. Solid Earth 120 (2015) 3142–3172.
[311] S. Munier, A. Polebistki, G. Belaud, D.P. Lettenmaier, SWOT data assimilation for operational reservoir management on the upper Niger River Basin, Wat. Res. Res. 51 (2015) 554–575.

[312] S.A. Murray, E.M. Henley, D.R. Jackson, S.L. Bruinsma, Assessing the performance of thermospheric modeling with data assimilation throughout solar cycles 23 and 24, Space Weather 13 (2015) 220–232.
[313] S.M. Naehr, F.R. Toffoletto, Radiation belt data assimilation with an extended Kalman filter, Space Weather 3 (2005) S06001.
[314] M.M. Najafabadi, F. Villanustre, T.M. Khoshgoftaar, N. Seliya, R. Wald, E. Muharemagic, Deep learning applications and challenges in big data analytics, J. Big Data 2 (2015) 1.
[315] S. Nakada, N. Hirose, T. Senjyu, K. Fukudome, T. Tsuji, N. Okei, Operational ocean prediction experiments for smart coastal fishing, Prog. Oceanogr. 121 (2014) 125–140.
[316] I.M. Navon, X. Zou, J. Derber, J. Sela, Variational data assimilation with an adiabatic version of the NMC spectral model, Mon. Wea. Rev. 120 (1992) 1433–1446.
[317] T.R. Nehrkorn, B.K. Woods, T. Aulignè, R.N. Hoffman, Application of feature calibration and alignment to high-resolution analysis: examples using observations sensitive to cloud and water vapor, Mon. Wea. Rev. 142 (2014) 686–702.
[318] T.R. Nehrkorn, B.K. Woods, R.N. Hoffman, T. Aulignè, Correcting for position errors in variational data assimilation, Mon. Wea. Rev. 143 (2015) 1368–1381.
[319] J.A.U. Nilsson, S. Dobricic, M. Pinardi, V. Taillandier, P.-M. Poulain, On the assessment of Argo float trajectory assimilation into the Mediterranean Forecasting System, Ocean Dyn. 61 (2011) 1475–1490.
[320] J. Nocedal, Updating quasi-Newton matrices with limited storage, Math. Comput. 38A (1986) 137–161.
[321] M. Nodet, Variational assimilation of Lagrangian data in oceanography, Inverse Probl. 22 (2006) 245–263.
[322] M. Nørgard, N.K. Poulson, O. Ravn, New developments in state estimation of nonlinear systems, Automatica 36 (2000) 1627–1638.
[323] A. Oba, T. Furumura, T. Maeda, Data assimilation-based early forecasting of long-period ground motions for large earthquakes along the Nankai Trench, J. Geophys. Res. Solid Earth 125 (2020) e2019JB019047.
[324] E. Ott, B.R. Hunt, I. Szunyogh, A.V. Zimin, E.J. Kostelich, M. Corazza, E. Kalnay, D.J. Patil, J.A. Yorke, A local ensemble transform Kalman filter for atmospheric data assimilation, Tellus 56A (2004) 415–428.
[325] T.N. Palmer, R. Gelero, J. Barkmeijer, R. Buizza, Singular vectors, metrics and adaptive observations, J. Atmos. Sci. 55 (1998) 633–653.
[326] C. Pan, L. Zheng, R.H. Weisberg, Y. Liu, C.E. Lembke, Comparisons of different ensemble schemes for glider data assimilation on West Florida Shelf, Ocean Model. 81 (2014) 12–24.
[327] N. Papadakis, E. Nemin, A. Cuzol, N. Gengembre, Data assimilation with the weighted ensemble Kalman filter, Tellus A 62 (2010) 673–697.
[328] D.F. Parrish, J.C. Derber, The National Meteorological Center's spectral statistical-interpolation analysis system, Mon. Wea. Rev. 120 (1992) 1747–1763.
[329] M.E. Pasyanos, G.A. Franz, A.L. Ramirez, Reconciling a geophysical model to data using a Markov chain Monte Carlo algorithm: an application to the Yellow Sea Korean Peninsula region, J. Geophys. Res. 111 (2006) B03313.
[330] T.J. Payne, Rapid update cycling with delayed observations, Tellus A 69 (2017) 1409061.
[331] J. Pearl, Causality, Models, Reasoning and Inference, Cambridge University Press, New York, NY, USA, 2007.
[332] J. Pedlosky, Geophysical Fluid Dynamics, Springer, New York, 1987.
[333] S.G. Penny, D.W. Behringer, J.A. Carton, E. Kalnay, A hybrid global ocean data assimilation system at NCEP, Mon. Wea. Rev. 143 (2015) 4660–4677.
[334] M. Peyron, A. Fillion, S. Gürol, V. Marchais, S. Gratton, P. Boudier, G. Goret, Latent space data assimilation by using deep learning, Q. J. R. Meteor. Soc. 147 (2021) 3759–3777.
[335] D.T. Pham, Stochastic methods for sequential data assimilation in strongly nonlinear systems, Mon. Wea. Rev. 129 (2001) 1194–1207.
[336] N.A. Phillips, A coordinate system having some special advantages for numerical forecasting, J. Met. Soc. 14 (1957) 184–185.

[337] A. Pikovsky, A. Politi, Dynamic localization of Lyapunov vectors in spacetime chaos, Nonlinearity 11 (1998) 1049–1062.
[338] S. Polavarapu, S. Ren, Y. Rochen, D. Sankey, N. Ek, J. Koshyk, D. Tarasick, Data assimilation with the Canadian middle atmosphere model, Atmos. Ocean 43 (1) (2005) 77–100.
[339] S. Polavarapu, M. Tanguay, R. Ménard, A. Staniforth, The tangent linear model for semi-Lagrangian schemes: linearizing the process of interpolation, Tellus 48A (1996) 74–95.
[340] P. Poli, H. Hersbach, D.P. Dee, P. Berrisford, A.J. Simmons, F. Vitart, P. Laloyaux, D.G.H. Tan, C. Peubey, J.-N. Thépaut, Y. Trémolet, E.V. Hólm, M. Bonavita, L. Isaksen, M. Fisher, ERA-20C: an atmospheric reanalysis of the twentieth century, J. Clim. 29 (2016) 4083–4097.
[341] R. Polkinghorne, T. Vukicevic, Data assimilation of cloud-affected radiances in a cloud-resolving model, Mon. Wea. Rev. 139 (2011) 755–773.
[342] D.J. Posselt, A Bayesian examination of deep convective squall-line sensitivity to changes in cloud microphysical parameters, J. Atmos. Sci. 73 (2016) 637–665.
[343] D.J. Posselt, D. Hodyss, C.H. Bishop, Errors in ensemble Kalman smoother estimates of cloud microphysical parameters, Mon. Wea. Rev. 142 (2014) 1631–1654.
[344] D.J. Posselt, T.S. L'Ecuyer, G.L. Stephens, Exploring the error characteristics of thin ice cloud property retrievals using a Markov chain Monte Carlo algorithm, J. Geophys. Res. 113 (2008) D24206.
[345] J. Poterjoy, A localized particle filter for high-dimensional nonlinear systems, Mon. Wea. Rev. 144 (2016) 59–76.
[346] J. Poterjoy, L. Wicker, M. Buehner, Progress towards the application of a localized particle filter for numerical weather prediction, Mon. Wea. Rev. 231 (2019) 1107–1126.
[347] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes, second ed., Cambridge University Press, New York, 1992.
[348] A. Priestly, A quasi-conservative version of the semi-Lagrangian scheme, Mon. Wea. Rev. 121 (1993) 621–629.
[349] N.C. Prive, R.M. Errico, Uncertainty of observation impact estimation in an adjoint model investigated with an Observing System Simulation Experiment, Mon. Wea. Rev. 147 (2019) 3191–3204.
[350] N.C. Privé, R.M. Errico, W. McCarty, The importance of simulated errors in observing system simulation experiments, Tellus A 73 (2021) 1–17.
[351] N.C. Prive, R.M. Errico, K.-S. Tai, Validation of forecast skill of the Global Modeling and Assimilation Office observing system simulation experiment, Q. J. R. Meteor. Soc. 139 (2013) 1354–1363.
[352] N.C. Prive, R.M. Errico, R. Todling, A. El Akkraoui, Evaluation of adjoint-based observation impacts as a function of forecast length using an Observing System Simulation Experiment, Q. J. R. Meteor. Soc. 147 (2021) 121–138.
[353] R.J. Purser, Accurate numerical differencing near a polar singularity of a skipped grid, Mon. Wea. Rev. 116 (1988) 1067–1076.
[354] R.J. Purser, L.M. Leslie, A semi-implicit semi-Lagrangian finite difference scheme using high-order spatial differencing on a non-staggered grid, Mon. Wea. Rev. 116 (1988) 2069–2080.
[355] C. Quilodrán Casas, R. Arcucci, P. Wu, C. Pain, Y.-K. Guo, A reduced order deep data assimilation model, Physica D 412 (2020) 132615.
[356] F. Rabier, P. Courtier, Four-dimensional assimilation in the presence of baroclinic instability. Nonlinear effects in 4D-Var, Q. J. R. Meteor. Soc. 118 (1992) 649–672.
[357] F. Rabier, H. Jarvinen, E. Klinker, J.-F. Mahfouf, A. Simmons, The ECMWF implementation of four dimensional variational assimilation. Part I: Experimental results with simplified physics, Q. J. R. Meteor. Soc. 126A (2000) 1143–1170.
[358] F. Rabier, E. Klinker, P. Courtier, A. Hollingsworth, Sensitivity of forecast errors to initial conditions, Q. J. R. Meteor. Soc. 122 (1996) 121–150.

[359] A. Rafieeinasab, D.-J. Seo, H. Lee, S. Kim, Comparative evaluation of maximum likelihood ensemble filter and ensemble Kalman filter for real-time assimilation of streamflow data into operational hydrologic models, J. Hydrol. 519 (2014) 2663–2675.
[360] A.L. Ramirez, J.J. Niato, W.G. Hanley, R. Aines, R.E. Glaser, S.K. Sengupta, K.M. Dyer, T.L. Hickling, W.D. Daily, Stochastic, inversion of electrical resistivity changes using a Marko chain Monte Carlo approach, J. Geophys. Res. 110 (2005) B02101.
[361] F. Rawlins, S.P. Ballard, K.J. Bovis, A.M. Clayton, D. Li, G.W. Inverarity, A.C. Lorenc, T.J. Payne, The Met Office global four-dimensional variational data assimilation scheme, Q. J. R. Meteor. Soc. 133 (2007) 347–362.
[362] L. Rayleigh, On convective currents in a horizontal layer of fluid when the higher temperature is on the underside, Phil. Mag. 32 (1916) 529–546.
[363] S. Reich, A nonparametric ensemble transform method for Bayesian inference, SIAM J. Sci. Comput. 35 (2013) A2013–A2024.
[364] L. Ren, M. Hartnett, S. Nash, Sensitivity tests of direct insertion data assimilation with pseudo measurements, Int. J. Comput. Commun. Eng. 13 (2014) 460–463.
[365] R.W. Reynolds, T.M. Smith, Improved global sea surface temperature analyses using optimum interpolation, J. Clim. 7 (1994) 929–948.
[366] L. Richardson, Weather Prediction by Numerical Process, University Press, Cambridge, MA, 1922.
[367] H. Ritchie, Eliminating the interpolation associated with the semi-Lagrangian scheme, Mon. Wea. Rev. 114 (1986) 135–146.
[368] A. Robert, A stable numerical integration scheme for the primitive meteorological equations, Atmos. Ocean 19 (1981) 35–46.
[369] T. Rosmond, L. Xu, Development of NAVDAS-AR: non-linear formulation and outer loop tests, Tellus 53A (2006) 45–58.
[370] T.E. Rosmond, A technical description of the NRL adjoint modeling system, Tech. report, Naval Research Laboratory, Monterey, CA, 1997.
[371] L.P. Rushøjgaard, S.E. Cohn, Y. Li, R. Ménard, The use of spline interpolation in semi-Lagrangian transport models, Mon. Wea. Rev. 126 (1998) 2008–2016.
[372] R. Sadourny, Conservative finite difference approximations of the primitive equations on quasi uniform spherical grids, Mon. Wea. Rev. 100 (1972) 136–144.
[373] R. Sadourny, A. Arakawa, A. Mintz, Integration of the non-divergent barotropic vorticity equation with an icosahedral-hexagonal grid for the sphere, Mon. Wea. Rev. 96 (1968) 351–356.
[374] H. Salman, C.K.R.T. Jones, K. Ide, A method for assimilating Lagrangian data into a shallow-water-equation ocean model, Mon. Wea. Rev. 134 (2006) 1081–1101.
[375] R. Salmon, Practical use of the Hamilton's principle, J. Fluid Mech. 132 (1982) 431–444.
[376] R. Salmon, New equations for the nearly geostrophic flow, J. Fluid Mech. 153 (1985) 461–477.
[377] R. Salmon, Semi-geostrophic theory as a Dirac-bracket projection, J. Fluid Mech. 196 (1988) 345–358.
[378] R. Salmon, Hamiltonian fluid dynamics, Annu. Rev. Fluid Dyn. 20 (1988) 225–256.
[379] R. Salmon, Lectures on Geophysical Fluid Dynamics, Oxford University Press, New York, 1998.
[380] B. Saltzman, Finite amplitude free convection as an initial value problem, J. Atmos. Sci. 19 (1962) 329–341.
[381] M. Sambridge, K. Mosengaard, Monte Carlo methods in geophysical inverse problems, Rev. Geophys. 40 (2002) G000089.
[382] J.A. Santanello Jr., S.V. Kumar, C.D. Peters-Lidard, P.M. Lawston, Impact of soil moisture assimilation on land surface model spinup and coupled land-atmosphere prediction, J. Hydromet. 17 (2016) 517–540.
[383] Y. Sasaki, A fundamental study of the numerical prediction based on the variational principle, J. Meteor. Soc. Japan 33 (1955) 262–275.
[384] Y. Sasaki, An objective analysis based upon variational methods, J. Meteor. Soc. Japan 36 (1958) 77–88.
[385] Y. Sasaki, Proposed inclusion of time variation terms, observational and theoretical, in numerical variational objective analysis, J. Meteor. Soc. Japan 47 (1969) 115–124.

[386] Y. Sasaki, Some basic formalisms in numerical variational analysis, Mon. Wea. Rev. 98 (1970) 875–883.
[387] Y. Sasaki, Numerical variational analysis formulated under the constraints as determined by longwave equations and a low-pass filter, Mon. Wea. Rev. 98 (1970) 884–899.
[388] Y. Sasaki, Numerical variational analysis with weak constraint and application to surface analysis of severe storm gust, Mon. Wea. Rev. 98 (1970) 900–912.
[389] Y. Sasaki, Some basic formalisms in numerical weather prediction, Mon. Wea. Rev. 98 (1970) 875–883.
[390] A.E. Schuh, T. Lauvaux, T.O. West, A.S. Denning, K.J. Davis, N. Miles, S. Richardson, M. Uliasz, E. Lokupitiya, D. Cooley, A. Andrews, S. Ogle, Evaluating atmospheric CO_2 invsersions at multiple scales over a highly inventoried agricultural landscape, Glob. Chang. Biol. 19 (2013) 1424–1439.
[391] C.J. Seaman, M. Sengupta, T.H. Vonder Haar, Mesoscale satellite data assimilation: impact of cloud affected infrared observations on a cloud-free initial state, Tellus 62A (2010) 298–318.
[392] F. Shen, J. Min, D. Xu, Assimilation of radar radial velocity data with the WRF hybrid ETKF-3DVAR system for the prediction of hurricane Ike, Atmos. Res. 169 (2016) 127–138.
[393] A. Shlyaeva, A. Buehner, M. Caya, J.-F. Lemieux, F. Smith, G. Roy, F. Dupont, T. Carrieres, Towards ensemble data assimilation for the Environment Canada Regional Ice Prediction System, Q. J. R. Meteor. Soc. 142 (2016) 1090–1099.
[394] A. Shrestha, A. Mahmood, Review of deep learning algorithms and architectures, IEEE Access 7 (2019) 53050–53065.
[395] I. Silberman, Planetary waves in the atmosphere, J. Meteor. 11 (1954) 27–34.
[396] L. Silvinski, E. Spiller, A. Apte, B. Sandstede, A hybrid particle-ensemble Kalman filter for Lagrangian data assimilation, Mon. Wea. Rev. 143 (2015) 195–211.
[397] T.C. Sluka, S.G. Penny, E. Kalnay, T. Mioshi, Assimilating atmospheric observations into the ocean using strongly coupled ensemble data assimilation, Geophys. Res. Lett. 43 (2016) 752–759.
[398] C.A. Smith, G.P. Compo, D.K. Hooper, Web-based reanalysis intercomparison tools (WRIT) for analysis and comparison of reanalyses and other datasets, Bull. Am. Meteor. Soc. (2014) 1671–1678.
[399] C.J. Smith, The semi-Lagrangian Method for Atmospheric Modelling, Ph.D. thesis, University of Reading, Department of Mathematics, 2000.
[400] G. Smith, F. Roy, M. Reszka, D.S. Colan, Z. He, D. Deacu, J.-M. Belanger, S. Skachko, Y. Liu, F. Dupont, J.-F. Lemieux, C. Beaudoin, B. Tranchant, M. Drévillon, G. Garric, C.-E. Testut, J.-M. Lellouche, P. Pellerin, H. Ritchie, Y. Lu, F. Davidson, M. Buehner, A. Caya, M. Lajoie, Sea ice forecast verification in the Canadian Global Ice Ocean Prediction System, Q. J. R. Meteor. Soc. 142 (2016) 659–671.
[401] P.J. Smith, A.M. Fowler, A.S. Lawless, Exploring strategies for coupled 4D-Var data assimilation using an idealised atmosphere-ocean model, Tellus 67 (2015) 27025.
[402] E.J. Smyth, M.S. Rayleigh, E.E. Small, Particle filter data assimilation of monthly snow depth observations improves estimation of snow density and SWE, Wat. Res. Res. 55 (2019) 1296–1311.
[403] C. Snyder, Particle filters, the "optimal" proposal and high-dimensional systems, in: Annual Seminar: Data Assimilation for Atmosphere and Ocean, ECMWF, 2011, pp. 161–170.
[404] C. Snyder, T. Bengtsson, P. Bickel, J. Anderson, Obstacles to high-dimensional particle filtering, Mon. Wea. Rev. 136 (2008) 4629–4640.
[405] C. Snyder, T. Bentsson, M. Morzfeld, Performance bounds for particle filter using the optimal proposal, Mon. Wea. Rev. 143 (2015) 4750–4761.
[406] C. Snyder, W.C. Skamarock, R. Rotunno, A comparison of primitive-equation and semigeostrophic simulations of baroclinic waves, J. Atmos. Sci. 48 (1991) 2179–2194.
[407] J.E. Solbrig, T.E. Lee, S.D. Miller, Advances in remote sensing: imagining the Earth by moonlight, EOS 94 (2013) 349–350.
[408] D. Solomentsev, K.S. Stanley, B. Khattatov, V. Khattatov, Y. Cherniak, A. Titov, Ionosphere data assimilation capabilities for representing the high-latitude geomagnetic storm event in September 2011, Space Weather 12 (2014) 10581–10594.

[409] H. Song, C.A. Edwards, A.M. Moore, J. Fiechter, Incremental four-dimensional variational data assimilation of positive-definite oceanic variables using a logarithm transformation, Ocean Model. 54 (2012) 1–17.
[410] M.R. Spiegel, Vector Analysis, Schaum's Outline, McGraw Hill, New York, 1959.
[411] A. Staniforth, J. Côté, Semi-Lagrangian integration schemes for atmospheric models: a review, Mon. Wea. Rev. 119 (1991) 2206–2223.
[412] P.G. Stegmann, B. Johnson, I. Moradi, B. Karpowicz, W. McCarty, A deep learning approach for fast radiative transfer, J. Quant. Spectrosc. Radiat. Transf. 280 (2022) 108088.
[413] R. Steinacker, D. Mayer, A. Steiner, Data quality control based on self-consistency, Mon. Wea. Rev. 139 (2011) 3974–3991.
[414] L.M. Stewart, S.L. Dance, N.K. Nichols, J.R. Eyre, J. Cameron, Estimating interchannel observation-error correlations for IASI radiance data in the Met Office system, Q. J. R. Meteor. Soc. 140 (2013) 1236–1244.
[415] A. Storto, G. De Magistris, S. Falchetti, P. Oddo, A neural network-based observation operator for coupled ocean-acoustic variational data assimilation, Mon. Wea. Rev. 149 (2021) 1967–1985.
[416] G. Strang, Linear Algebra and Its Applications, second ed., Academic Press, New York, USA, 1980.
[417] G. Strang, G. Fix, An Analysis of the Finite Element Method, Prentice Hall, Englewood-Cliffs, NJ, 1973.
[418] J.R. Stroud, B.M. Lesht, D.J. Schwab, D. Beletsky, M.L. Stein, Assimilation of satellite images into a sediment transport model of Lake Michigan, Wat. Res. Res. 45 (2009) W02419.
[419] L. Sun, S.G. Penny, Lagrangian data assimilation of surface drifters in a double-gyre ocean model using the local ensemble transform Kalman filter, Mon. Wea. Rev. 147 (2019) 4533–4551.
[420] K. Sung, H. Song, I. Kwon, A local unscented transform Kalman filter, Mon. Wea. Rev. 148 (2020) 3243–3266.
[421] V. Taillandier, A. Griffa, A. Molcard, A variational approach for the reconstruction of regional scale Eulerian velocity fields from Lagrangian data, Ocean Model. 13 (2006) 1–24.
[422] O. Talagrand, Four-dimensional variational data assimilation, in: ECMWF Seminar Proceedings on Data Assimilation and the Use of Satellite Data, 1988, pp. 1–30.
[423] S.K. Tamang, A. Ebtehaj, D. Zou, G. Lerman, Regularized variational data assimilation for bias treatment using the Wasserstein metric, Q. J. R. Meteor. Soc. 146 (2020) 2332–2436.
[424] J. Tamminen, Validation of nonlinear inverse algorithms with Markov chain Monte Carlo method, J. Geophys. Res. 109 (2004) D19303.
[425] J. Tamminen, E. Kyrölä, Bayesian solution for nonlinear and non-Gaussian inverse problems by Markov chain Monte Carlo method, J. Geophys. Res. 106 (2001) 14377–14390.
[426] Q. Tang, R. Hobbs, C. Zheng, B. Biecscas, C. Caiado, Marko chain Monte Carlo inversion of temperature and salinity structure of an internal solitary wave packet from marine seismic data, J. Geophys. Res. 121 (2016) C011810.
[427] M. Tanguay, S. Polavarapu, The adjoint of the semi-Lagrangian treatment of the passive tracer equation, Mon. Wea. Rev. 127 (1999) 551–564.
[428] M. Tanguay, E. Yakimiw, H. Ritchie, A. Robert, Advantages of spatial averaging in semi-implicit and semi-Lagrangian schemes, Mon. Wea. Rev. 120 (1992) 113–123.
[429] A. Tarantola, Inverse Problem Theory and Methods for Model Parameter Estimation, SIAM, Philadelphia, 2005.
[430] A. Tarantola, B. Valette, Inverse problems = quest for information, J. Geophys. 50 (1982) 159–170.
[431] A. Tarantola, B. Valette, Generalized nonlinear inverse problems solved using the least squares criterion, Rev. Geophys. Space Phys. 20 (1982) 219–232.
[432] C. Tavolato, L. Isaksen, On the use of the Huber norm for observation quality control in the ECMWF 4D-Var, Q. J. R. Meteor. Soc. 141 (2015) 1514–1527.
[433] C. Temperton, Implicit normal model initialization, Mon. Wea. Rev. 116 (1988) 1013–1031.
[434] C. Temperton, A. Staniforth, An efficient two-time-level semi-Lagrangian semi-implicit integration scheme, Q. J. R. Meteor. Soc. 115 (1987) 1025–1039.

[435] J.-N. Thépaut, P. Courtier, Four-dimensional variational data assimilation using the adjoint of a multilevel primitive equation model, Q. J. R. Meteor. Soc. 117 (1991) 1225–1254.
[436] J.-N. Thépaut, R.N. Hoffman, P. Courtier, Interactions of dynamics and observations in four-dimensional variational assimilation, Mon. Wea. Rev. 121 (1993) 3393–3414.
[437] H.J. Thiebaux, Experiments with correlation representations for objective analysis, Mon. Wea. Rev. 103 (1975) 617–627.
[438] J. Thuburn, A PV-based shallow-water model on a hexagonal-icosahedral grid, Mon. Wea. Rev. 125 (1997) 2328–2347.
[439] M.K. Tippett, J.L. Anderson, C.H. Bishop, T.M. Hamil, J.S. Whitaker, Ensemble square root filters, Mon. Wea. Rev. 131 (2003) 1485–1490.
[440] Z. Toth, E. Kalnay, Ensemble forecasting at NMC: the generation of perturbations, Bull. Am. Meteor. Soc. 74 (1993) 2317–2330.
[441] Z. Toth, E. Kalnay, Ensemble forecasting at NCEP: the breeding method, Mon. Wea. Rev. 125 (1997) 3297–3318.
[442] Y. Trèmolet, Accounting for an imperfect model in 4D-Var, Q. J. R. Meteor. Soc. 132 (2006) 2483–2504.
[443] Y. Trèmolet, Model-error estimation in 4D-Var, Q. J. R. Meteor. Soc. 133 (2007) 1267–1280.
[444] P. Uhe, M. Thatcher, A spectral nudging method for the ACCESS1.3 atmospheric model, Geosci. Model Dev. 8 (2015) 1645–1658.
[445] B. Uzunoglu, S.J. Fletcher, M. Zupanski, I.M. Navon, Adaptive ensemble reduction and inflation, Q. J. R. Meteor. Soc. 133 (2007) 1281–1294.
[446] M.S. Van Den Broeke, D.M. Tobin, M.P. Kumjian, Polarimetric radar observations of precipitation type and rate from the 2–3 March 2014 Winter storm in Oklahoma and Arkansas, Wea. Forecast. 31 (2016) 1179–1196.
[447] R. Van der Merwe, E.A. Wan, Efficient derivative-free Kalman filters for online learning, in: Proc. 2001 European Symp. on Artificial Neural Networks (ESANN), Bruges, Belgium, 2001.
[448] R. Van der Merwe, E.A. Wan, The square-root unscented Kalman filter for state and parameter estimation, in: Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), Vol. 6, Salt Lake City, UT, IEEE, 2001, pp. 3461–3464.
[449] P.-J. Van Leeuwen, E. Evensen, Data assimilation and inverse methods in terms of probabilistic formulations, Mon. Wea. Rev. 124 (1996) 2898–2913.
[450] P.-J. Van Leeuwen, G. Evensen, Data assimilation and inverse methods in terms of probabilistic formulation, Mon. Wea. Rev. 124 (1996) 2898–2913.
[451] P.J. van Leeuwen, A variance-minimizing filter for large-scale applications, Mon. Wea. Rev. 131 (2003) 2071–2084.
[452] P.J. van Leeuwen, Particle filtering in geophysical systems, Mon. Wea. Rev. 137 (2009) 4089–4114.
[453] P.J. van Leeuwen, Nonlinear data assimilation in geosciences: an extremely efficient particle filter, Q. J. R. Meteor. Soc. 136 (2010) 1991–1999.
[454] P.J. van Leeuwen, H.R. Künscu, L. Nerger, R. Potthast, S. Reich, Particle filter for high-dimensional geoscience applications: a review, Q. J. R. Meteor. Soc. 145 (2019) 2335–2365.
[455] M. van Lier-Walqui, A.M. Fridlind, A.S. Ackerman, S. Collis, J. Helmus, D.R. MacGorman, K. North, P. Kollias, D.J. Posselt, On polarimetric radar signatures of deep convection for model evaluation: columns of specific differential phase observed during MC3E, Mon. Wea. Rev. 144 (2016) 737–758.
[456] M. van Lier-Walqui, T. Vukievic, D.J. Posselt, Quantification of cloud microphysics parametrization uncertainty using radar reflectivity, Mon. Wea. Rev. 140 (2012) 3442–3466.
[457] J. Vialard, A.T. Weaver, D.L.T. Anderson, P. Delecluse, Three and four-dimensional variational assimilation with a general circulation model of the tropical Pacific Ocean. Part II: Physical validation, Mon. Wea. Rev. 131 (2003) 1379–1395.
[458] G. Wahba, D.R. Johnson, F. Gao, J. Gong, Adaptive tuning of numerical weather prediction models: randomized GCV in three- and four-dimensional data assimilation, Mon. Wea. Rev. 123 (1995) 3358–3369.

[459] J.A. Waller, S.L. Dance, N.K. Nichols, Theoretical insight into diagnosing observation error correlations using observation-minus-background and observation-minus-analysis statistics, Q. J. R. Meteor. Soc. 141 (2016) 418–431.
[460] E.A. Wan, R. Van der Merwe, The unscented Kalman filter for nonlinear estimation, in: Proc. 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (ASSPCC), Lake Louise, AB, Canada, IEEE, 2000, pp. 153–158.
[461] J. Wang, V.R. Kotamarthi, Assessment of dynamical downscaling in the near surface fields with different spectral nudging approaches using the nested regional climate model (NRCM), J. Appl. Meteor. Clim. 52 (2013) 1576–1591.
[462] X. Wang, Incorporating ensemble covariance in the Gridpoint Statistical Interpolation variational minimization: a mathematical framework, Mon. Wea. Rev. 138 (2010) 2990–2995.
[463] X. Wang, C.H. Bishop, A comparison of breeding and ensemble transform Kalman filter ensemble forecast schemes, J. Atmos. Sci. 60 (2003) 1140–1158.
[464] X. Wang, C.H. Bishop, S.J. Jilier, Which is better, an ensemble of positive-negative pairs or a centered spherical simplex ensemble?, Mon. Wea. Rev. 132 (2004) 1590–1605.
[465] X. Wang, D. Parrish, D. Kleist, J. Whitaker, GSI 3DVar-based ensemble-variational hybrid data assimilation for NCEP Global Forecasting System: single-resolution experiments, Mon. Wea. Rev. 141 (2013) 4098–4117.
[466] Y. Wang, K. Satake, T. Maeda, A. Riadi Gusman, Green's Function-Based Tsunami Data Assimilation: a Fast data assimilation approach towards Tsunami early warning, Geophys. Res. Lett. 44 (2017) 10,282–10,289.
[467] J. Waters, D.J. Lea, M.J. Matrin, I. Mirouze, A. Weaver, J. While, Implementing a variational data assimilation system in an operational 1/4 degree global ocean model, Q. J. R. Meteor. Soc. 141 (2011) 755–773.
[468] A.T. Weaver, J. Vialard, D.L.T. Anderson, Three- and four-dimensional variational data assimilation with a general circulation model of the tropical Pacific Ocean. Part I: Formulation, internal diagnostics and consistency checks, Mon. Wea. Rev. 131 (2003) 1360–1378.
[469] P. Welander, Studies on the general development of motion in a two-dimensional ideal fluid, Tellus 17 (1955) 141–156.
[470] J.A. Weyn, D.R. Durrran, R. Caruana, N. Cresswell-Clay, Sub-seasonal forecasting with a large ensemble of Deep-Learning Weather Prediction Models, J. Adv. Model. Earth Syst. 13 (2021) e2021MS002502.
[471] J.S. Whitaker, G.P. Compo, An ensemble Kalman smoother for reanalysis, in: Proc. Symp. on Observations, Data Assimilation and Probabilistic Prediction, Orlando, FL, Amer. Meteor. Soc., 2002, pp. 144–147.
[472] J.S. Whitaker, T.M. Hamil, Ensemble data assimilation without perturbed observations, Mon. Wea. Rev. 130 (2002) 1913–1924.
[473] A.A. White, A view of the equations of meteorological dynamics and various approximations, in: J. Norbury, I. Roulstone (Eds.), Large-Scale Atmosphere-Ocean Dynamics, 1: Analytical Methods and Numerical Models, Cambridge University Press, Cambridge, 2002, pp. 1–100.
[474] J.S. Whittaker, T. Hamill, Evaluating methods to account for system errors in ensemble data assimilation, Mon. Wea. Rev. 140 (2012) 3078–3089.
[475] N. Wiener, The Extrapolation, Interpolation and Smoothing Stationary Time Series, John Wiley and Sons, Inc., New York, 1949.
[476] A. Wiin-Nielson, On the application of trajectory methods in numerical forecasting, Tellus 11 (1959) 180–196.
[477] D.L. Williamson, Integration of barotropic vorticity equations on a spherical geodesic grid, Tellus 20A (1968) 642–653.
[478] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, P.N. Swartrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, J. Comput. Phys. 102 (1992) 211–224.
[479] D.L. Williamson, P.J. Rasch, Two-dimensional semi-Lagrangian transport with shape-preserving interpolation, Mon. Wea. Rev. 117 (1989) 117–129.

[480] D.L. Williamson, P.J. Rasch, On slope preserving interpolation and semi-Lagrange transport, SIAM J. Sci. Stat. Comput. 11 (1990) 656–687.
[481] D.M. Winslow, A.T. Fisher, K. Becker, Characterizing borehole fluid flow and formation permeability in the ocean crust using linked analytic models and Marko chain Monte Carlo analysis, Geochem. Geophys. Geosyst. 14 (2013) 3857–3874.
[482] M.L. Wrzesien, S. Kumar, C. Vuyovich, E.D. Gutman, R. Sung Kim, B.A. Forman, M. Durrand, M.S. Raleigh, R. Webb, P. Houser, Development of a "Nature Run" for Observing System Simulation Experiments (OSSEs) for Snow Mission Development, J. Hydromet. 23 (2022) 351–375.
[483] C. Xia, C. Cochrane, J. DeGuire, G. Fan, E. Holmes, M. McGuirl, P. Murphy, J. Palmer, P. Carter, L. Slivinski, B. Sandstede, Assimilation of Eulerian and Lagrangian data in traffic-flow models, Physica D 346 (2017) 59–72.
[484] Q. Xiao, J. Sun, Multiple radar data assimilation and short range quantitative precipitation forecasting of a squall line observed during IHOP_2002, Mon. Wea. Rev. 135 (2007) 3381–3404.
[485] F. Xu, A. Ignatov, In situ SST Quality Monitor (iQuam), J. Atmos. Ocean. Tech. 31 (2014) 164–180.
[486] J. Xu, H. Shu, Assimilating MODIS based albedo and snow cover fraction into the Common Land Model to improve snow depth simulation with direct insertion and deterministic ensemble Kalman filter methods, J. Geophys. Res. Atmos. 119 (2014) 10684–10701.
[487] L. Xu, R. Daley, Towards a true 4-dimensional data assimilation algorithm: application of a cycling representer algorithm to a simple transport problem, Tellus 52A (2000) 109–128.
[488] L. Xu, R. Daley, Data assimilation with a barotropically unstable shallow water system using representer algorithms, Tellus 54A (2002) 125–137.
[489] L. Xu, T. Rosmond, R. Daley, Development of NAVDAS-AR: formulation and initial test of the linear problem, Tellus 58A (2005) 546–559.
[490] Z. Xu, Z.-L. Yang, A new dynamical downscaling approach with GCM bias correction and spectral nudging, J. Geophys. Res. 120 (2015) 3063–3084.
[491] M. Xue, M. Hu, A.D. Schenkman, Numerical prediction of the 8 May 2003 Oklahoma City tornadic supercell and embedded tornado using ARPS with the assimilation of WSR-88D data, Wea. Forecast. 29 (2014) 39–62.
[492] A. Yamazaki, T. Miyoshi, J. Inoue, T. Enomoto, N. Komori, EFSO at different geographical locations verified with observing system experiments, Wea. Forecast. 36 (2021) 1219–1236.
[493] C. Yang, Z. Liu, J. Bresch, S.R.H. Rizvi, X.-Y. Haung, J. Min, AMSR2 all-sky radiance assimilation and its impact on the analysis and forecast of Hurricane Sandy with a limited-area data assimilation system, Tellus 68 (2016) 30917.
[494] J.R. Yearsley, A semi-Lagrangian water temperature model for advection-dominated river systems, Wat. Res. Res. 45 (2009) 1–19.
[495] X. Yin, T. Dai, N.A.J. Schutgens, D. Goto, T. Nakajima, G. Shi, Effects of data assimilation on the global aerosol key optical properties simulations, Atmos. Res. 178–179 (2016) 175–186.
[496] X. Yue, W.S. Schreiner, N. Pedatella, R.A. Anthes, A.J. Mannucci, P.R. Strauss, J.-Y. Liu, Space weather observations by GNSS radio occultation: from FORMOSAT-3/COSMIC to FORMOSAT-7/COSMIC-2, Space Weather 12 (2014) 616–621.
[497] J. Zhang, J.R. Campbell, E.J. Hyer, J.S. Reid, D.L. Westphal, R.S. Johnson, Evaluating the impact of multisensor data assimilation a global aerosol particle transport model, J. Geophys. Res. 119 (2014) 4674–4689.
[498] W.G. Zhang, J.L. Wilkin, H.G. Arango, Towards and integrated observation and modeling system in the New York Bight using variational methods. Part I: 4DVAR data assimilation, Ocean Model. 35 (2010) 119–133.
[499] X. Zhang, Y. Luo, Q. Wan, W. Ding, J. Sun, Impact of assimilating wind profiling radar observations on convection-permitting quantitative precipitation forecasts during SCMREX, Wea. Forecast. 31 (2016) 1271–1292.
[500] Y. Zhao, S.J. Breybush, R.J. Wilson, R.N. Hoffman, E. Kalnay, Impact of assimilation window length on diurnal features in a Mars atmospheric analysis, Tellus 67A (2015) 26042.

[501] Y. Zhu, J.C. Derber, R.J. Purser, B.A. Ballish, J. Whitting, Variational correction of aircraft temperature bias in the NCEP's GSI analysis system, Mon. Wea. Rev. 143 (2015) 3774–3803.
[502] C. Zoccarato, D. Baù, M. Ferronato, G. Gambolati, A. Alzraiee, P. Teatini, Data assimilation of surface displacements to improve geomechanical parameters of gas storage reservoirs, J. Geophys. Res. Solid Earth 121 (2016) 1441–1461.
[503] D. Zupanski, A general weak constraint applicable to operational 4DVAR data assimilation systems, Mon. Wea. Rev. 125 (1997) 2274–2292.
[504] D. Zupanski, A.Y. Hou, S.Q. Zhang, M. Zupanski, C.D. Kummerow, S.H. Cheung, Applications of information theory in ensemble space, Q. J. R. Meteor. Soc. 133 (2007) 1533–1545.
[505] M. Zupanski, Regional four-dimensional variational data assimilation in a quasi-operational forecasting environment, Mon. Wea. Rev. 121 (1993) 2396–2408.
[506] M. Zupanski, A preconditioning algorithm for large-scale minimization problems, Tellus 45A (1993) 478–492.
[507] M. Zupanski, Maximum likelihood ensemble filter. Part I: Theoretical aspects, Mon. Wea. Rev. 133 (2005) 1710–1726.
[508] M. Zupanski, The Maximum Likelihood Ensemble Filter with state space localization, Mon. Wea. Rev. 149 (2021) 3505–3524.
[509] M. Zupanski, S.J. Fletcher, I.M. Navon, B. Uzunoglu, R.P. Heikes, D.A. Randall, T.D. Ringler, D. Deascu, A method for initiation of ensemble data assimilation, Tellus 58A (2006) 159–170.
[510] M. Zupanski, I.M. Navon, D. Zupanski, The maximum likelihood ensemble filter as a non-differentiable minimization algorithm, Q. J. R. Meteor. Soc. 134 (2008) 1039–1050.
[511] M. Zupanski, D. Zupanski, T. Vukicevic, K. Eis, T. Vonder Harr, CIRA/CSU four-dimensional variational data assimilation system, Mon. Wea. Rev. 133 (2005) 829–843.

Index

C

D

E

F

G

J

K

L

M

P

Q

R

S

T

W

Y

Z

SECOND EDITION

DATA ASSIMILATION FOR THE GEOSCIENCES

FROM THEORY TO APPLICATION

Authored by
Steven J. Fletcher

Revised edition covering the mathematics, statistics and probability theory needed to understand data assimilation, written with geoscientists in mind

Data Assimilation for the Geosciences: From Theory to Application, Second Edition brings together all the mathematical and statistical background knowledge needed to formulate data assimilation systems into one place. It includes practical exercises enabling readers to apply theory in both a theoretical formulation as well as teach them how to code the theory with toy problems to verify their understanding. It also demonstrates how data assimilation systems are implemented in larger scale fluid dynamical problems related to land surface, the atmosphere, ocean, and other geophysical situations.

The second edition of ***Data Assimilation for the Geosciences*** has been revised with up-to-date research in data assimilation, as well as how to apply the techniques. The new edition features an introduction of how artificial intelligence is interfacing and aiding data assimilation. In addition to appealing to students and researchers across the geosciences, this edition also appeals to new students and scientists in the field of data assimilation due to the updates and more information on techniques, research, and applications, consolidated into one source.

Key Features

- Includes practical exercises and solutions enabling readers to apply theory in both a theoretical formulation as well as enabling them to code theory
- Provides the mathematical and statistical background knowledge needed to formulate data assimilation systems into one place
- Presents of new topics such as Observing System Experiments (OSE) and Observing System Simulation Experiments (OSSE), Lagrangian data assimilation, and artificial intelligence and data assimilation

About the Author

Steven J. Fletcher is a Research Scientist III at the Cooperative Institute for Research in the Atmosphere (CIRA) at Colorado State University, where he is the lead scientist on the development of non-Gaussian based data assimilation theory for variational, PSAS, and hybrid systems. He has worked extensively with the Naval Research Laboratory in Monterey in development of their data assimilation system, as well as working with the National Atmospheric and Oceanic Administration (NOAA)'s Environmental Prediction Centers (EMC) data assimilation system. Dr. Fletcher is extensively involved with the American Geophysical Union (AGU)'s Fall meeting planning committee, having served on the committee since 2013 as the representative of the Nonlinear Geophysics section. He has also been the lead organizer and science program committee member for the Joint Center for Satellite Data Assimilation Summer Colloquium on Satellite Data Assimilation since 2016. In 2017 Dr. Fletcher became a fellow of the Royal Meteorological Society.

ISBN 978-0-323-91720-9

ELSEVIER elsevier.com/books-and-journals