THE UNIVERSITY OF READING

DEPARTMENT OF MATHEMATICS

Data Assimilation for Numerical Weather Prediction Using Control Theory

by

Anne K. Griffith

Thesis submitted for the degree of

Doctor of Philosophy

April 1997

Abstract

Data assimilation is a means of estimating an atmospheric or oceanic state by combining observational data with a prior estimate of the state, usually from a numerical model. We look at application of data assimilation to numerical weather prediction using control theory.

Firstly, we apply observer theory to successive correction methods of data assimilation to show when they converge in time to the true solution. However, we mostly focus on 4D variational data assimilation schemes. Here the approach is to minimize a cost function penalizing distance from observational data over a time interval, subject to the constraint that the model equations are satisfied. The minimization problem can be solved by iterating on the model initial state, which is referred to as "using the initial state as the control vector".

Our aim is to provide a consistent theoretical foundation which allows for model error in variational assimilation. We investigate the "correction term technique" in which a constant correction term approximating model error is added to the model equations and used as a control vector instead of, or as well as, the initial state. We use the concept of complete N-step observability to give conditions for a unique solution of the minimization problem using different control vectors.

We suggest a generalization of the correction term technique in which we use state augmentation to estimate a serially correlated component of model error along with the model state. In particular, we consider using a correction term representing model error that evolves as the model state evolves. We investigate the effectiveness of the constant and the evolving correction term in compensating for different types of model error using simple linear models. We also use the correction term technique for a 1D nonlinear shallow water model in the presence of different types of model error, and find that a constant correction term can compensate for non-constant model error on a significant timescale.

Acknowledgements

First of all, I thank my academic supervisor, Prof. Nancy Nichols, for what I have learnt from her about mathematics and about doing research, and I thank her for further inspiring my interest in the subject. I also thank my industrial supervisor, Mr Andrew Lorenc of the U.K. Meteorological Office, for a chance to learn about data assimilation at the Met. Office, and for valuable discussions about this work.

I would like to thank my family and my friends who encouraged me in my work. I especially thank my "chocaholic" office mates, Alison, Chris and Lee, who helped to make my time as a PhD student particularly enjoyable.

Finally, I acknowledge financial support for my studies from the Engineering and Physical Sciences Research Council and from the U.K. Meteorological Office.

Contents

Chapter 1

Introduction

We start with an introduction to data assimilation, particularly focusing on its application to numerical weather prediction (NWP). This is followed by an overview of the rest of the thesis.

1.1 Background on data assimilation

In meteorology and oceanography, *data assimilation* is a means of estimating the state of the atmosphere or ocean by combining *observational data* with a *prior estimate* of the state, which usually comes from a dynamical model. This estimate of the atmospheric or oceanic state is often called an *analysis*.

Three important applications of data assimilation are: to provide a good analysis of the current situation to be used in initiating a forecast; to give a good analysis of a past event for diagnostic studies or archive records; and to use observational data for the process of model verification and for increasing our knowledge of physical processes.

In meteorology, the main application of data assimilation is in NWP, where it is used to obtain a good estimate of the current atmospheric state for initiating a forecast. Typically, data in a time window of 10 or 12 hours is assimilated to give an analysis of the "current" state, to be used as initial conditions for a forecast. It is this application that we refer to as *operational data assimilation*. Data assimilation is also used in a non-operational context for diagnostic studies of the atmosphere,

in forecast verification, for archive records and for climate studies. In oceanography, on the other hand, the main use of data assimilation is in studies to increase understanding of ocean circulations, although it is also used in short range ocean forecasting [32].

A wide variety of data assimilation schemes have been proposed and developed over the last 50 years or so, and many of them are taken from state estimation techniques in engineering. We now introduce the terminology we will use to describe different types of data assimilation schemes. By *three dimensional* (3D) data assimilation schemes, we mean schemes which are designed to provide an analysis at a single time, and treat each analysis time in isolation. In contrast to 3D schemes, *four dimensional* (4D) schemes seek to benefit from the "time-tendency information" in the observations. In 4D schemes, information from observations at earlier (and in some cases later) times is used in the analysis at a given time. 4D data assimilation schemes involve a model of the state evolution.

Sequential data assimilation schemes treat observations as they occur in time, and then discard them [32]. If a 3D analysis is carried out repeatedly, this can be seen as a sequential approach to data assimilation, and hence 3D schemes can be regarded as sequential methods of data assimilation. 4D sequential schemes seek to find an analysis which draws closer to the true solution as time progresses, and as more information from observations becomes available. In these schemes information from observations at earlier times influences the solution at any time. In the control theory literature, 4D sequential data assimilation is referred to as *filtering* [44].

The *4D variational assimilation* schemes, on the other hand, use information from observations at both earlier and later times in a given *assimilation interval* for an analysis at any given time in the assimilation interval. For a set of observations over a given assimilation interval, a 4D sequential data assimilation scheme is designed to give the "best possible analysis" (in some sense) at the end of the time interval, and a 4D variational scheme is designed to give the best possible analysis over the entire assimilation interval.

4D variational assimilation can be expressed as a constrained minimization problem. The aim is to minimize a cost function penalizing distance from the observa-

tions over the assimilation interval, and distance from a prior estimate, subject to the constraint that the solution (the analysis) is consistent with the model dynamics. In the *strong constraint* approach to the 4D variational assimilation problem, the constraint is that the solution must satisfy the model equations exactly. In the *weak constraint* approach, the solution is only required to satisfy the model equations approximately, and hence some allowance is made for *model error*.

Meteorological observational data available for assimilation

We now give a brief description of the types of observational data available for use in operational meteorological assimilation to produce analyses for weather forecasting, a fuller description is given in the book by Daley [24].

Meteorological observations are available on a world-wide scale, and there is international cooperation on the data collection and distribution to the various national meteorological centres. At surface level, observations are available from land weather stations and over the sea from ships. These observations are available at least every three hours, usually at the sub-synoptic times, ie 0000GMT, 0300GMT,... Radiosondes and pilot balloons are launched from land areas and from ships, and typically give observational data at the synoptic times, ie 0000GMT and 1200GMT.

An increasingly important source of data is that of reports from commercial aircraft. These reports provide an increase in the spatial coverage of data, and a more continuous temporal coverage, but are of course limited to the well-travelled routes. Satellite information provides greater global coverage, and also greater vertical coverage. This data is again available continuously in time (ie, is asynoptic). One important aspect of satellite data is that it typically is nonlinearly related to the model state variables. It is likely that the availability of satellite data will increase still further in the near future [60].

More detail on what meteorological variables are observed in each of these data sources and also the sizes of typical observational errors in each case are given in [24]. Observational data varies enormously in type and accuracy, and also in spatial and temporal distribution. This is an important point to consider in design of a data assimilation scheme.

The quality control of observations is crucial for successful data assimilation. Data needs to be checked for gross errors and for internal consistency. This may be done before data is assimilated, or as part of the data assimilation process itself. Generally, it is assumed that observational errors are Gaussian [58]. Information on observational error correlations and on how to specify the observation error covariance matrices needed in many applications of data assimilation can be found in [24].

Numerical models used in data assimilation

The type of numerical model used in data assimilation depends of course on the application. In the context of operational data assimilation in meteorology, the forecast model itself, or perhaps a simplified version of it, is used. Data assimilation is carried out on both global models and mesoscale limited area models in an operational context. The models used may be finite difference models, finite element models or spectral models. Current operational weather forecast models typically have a model state with dimension of the order of 10^5 to 10^7. This huge number of model unknowns at each timestep has a very dominant impact on the practical choice of a data assimilation scheme.

The models most accurately describing atmospheric or oceanic evolution are nonlinear. These models can exhibit chaotic behaviour, and this has also been observed in the laboratory for some types of flow [63]. However, nonlinearities in atmospheric and oceanic flows are essentially quadratic, and the nonlinear effects do not dominate on the time-scales of operational data assimilation, although they can have a huge impact on longer timescales, [32]. For mid-latitude atmospheric flows, Lacarra and Talagrand [48] have showed that the *tangent linear model* is a good approximation to the full nonlinear model for a period of about 48 hours.

Initialization

In numerical weather prediction (NWP), *initialization* is a process of reducing the inertia-gravity waves present at the beginning of a forecast as much as possible. This is necessary in the context of data assimilation with realistic models, because dis-

crepancy between noisy data and a prior estimate of the state can produce spurious inertia-gravity waves [32]. Although primitive equation models do in fact exhibit gravity waves which describe a small amount of the flow, atmospheric and oceanic flows at mid-latitudes on the timescale of a forecast are well described by the relatively slow Rossby waves.

Early NWP models were often quasi-geostrophic, and hence avoided the need for initialization, since these models produce only Rossby waves. When primitive equation models became operational for forecasting in the early 1970s, initialization became necessary. Initialization has generally been carried out separately from the data assimilation procedure, by projection of the solution onto the subspace described by the Rossby modes [32], or by the process of nonlinear normal mode initialization introduced by Machenhauer [59]. In "advanced" data assimilation techniques, however, it is possible to incorporate the initialization process in the assimilation. This may be done in Kalman filtering applications by projection of the solution onto the Rossby modes [32], and in variational assimilation applications by the addition of a penalty term to the cost function [19], [91], [94].

Brief historical overview of data assimilation methods

Here we mention briefly the main methods that have been used for data assimilation in meteorology and oceanography, and methods that are currently being developed. More detail on the methods themselves with more references on their application are given in Chapters 3 and 4.

In the 1940s and 1950s, along with the advent of primitive computers, interest grew in finding methods for *objective analysis* of the atmosphere. The earliest attempts involved using polynomial splines to fit the data. This was done by Panofsky in 1949 [68], and by Gilchrist and Cressman in 1954 [35]. The method of *successive corrections*, introduced by Bergthorsen and Döös in 1955 [9] and Cressman in 1959 [21], proved more appropriate when less dense data coverage was available, and variants of this method have been used successfully in operational data assimilation.

Schemes taking into account the relative accuracy of observations and corresponding prior estimates of the state from numerical models were proposed early

on, but only as computer power increased was this approach further developed and used extensively in an operational context. The method of *optimal interpolation* (OI) suggested by Gandin in 1963 [30], attempts to provide a statistically optimal estimate of a linear system at a given time. Variants of this method, which are also applicable to nonlinear systems, have been applied widely for operational data assimilation in the 1980s and 1990s. The *three-dimensional variational assimilation* (or 3DVAR) method [73] can be seen as a different approach to solving the same problem as OI, and is currently being developed for operational use at several NWP centres.

In the earlier days of data assimilation, observations were available mainly at the synoptic and sub-synoptic times. Since observations from satellites have become available, however, some observations are available continuously and it has become more important that data assimilation techniques should draw upon the time-tendency information available in the observations. For this reason, there is much interest at present in the design and development of 4D data assimilation methods. Two examples of such methods include the Kalman filter, and 4D variational assimilation.

The *Kalman filter*, proposed by Kalman in 1960 [45] for engineering applications can be used as a sequential 4D assimilation method. For a linear model and under certain assumptions, it provides a statistically optimal solution at a given time taking into account all previous observations. The method in unsimplified form is generally considered too expensive for use with large operational models in meteorology and oceanography [32], but various simplifications have been proposed which are feasible [84]. Kalman filtering theory can also be extended for use with nonlinear models.

The *four-dimensional variational assimilation* method was suggested for meteorological data assimilation by Sasaki in 1958 [75]. the method seeks to obtain an optimal solution over an entire assimilation interval by minimization of a cost function penalizing distance from the observations and from a prior estimate of the state. The minimization is subject to the constraint that the model equations hold, either exactly (the *strong constraint* approach), or approximately (the *weak constraint* approach). Using the strong constraint approach, the problem can be reduced to that

of finding the optimal initial state for the assimilation interval [51]. This approach to 4D variational assimilation has received much attention since the mid 1980s, and several meteorological centres are currently developing it for eventual operational implementation in the late 1990s.

Under certain statistical assumptions, the weak constraint formalism, which allows for model error, gives the same statistically optimal solution as the Kalman filter at the end of an assimilation interval. The problem of finding 4D variational assimilation methods that can account for model error at reasonable cost is a problem currently receiving attention in research.

1.2 Overview of the thesis

In Chapter 2 we present mathematical background useful for the methods of data assimilation we consider in this thesis. We include definitions and useful results from control theory, an overview of nonlinear optimization theory, background on descent methods and a brief overview of probability theory.

In Chapter 3, we look at sequential methods of data assimilation. We give some background on 3D data assimilation methods and on the Kalman filter. When describing the Kalman filter, we focus on the assumptions made on model error and observational error, and on how to allow for serially correlated model error, since we refer to these issues later. We then give background on observer theory, and describe as an example of observer design, a *robust observer*. We note that observer theory is useful for data assimilation and that 4D sequential data assimilation methods such as the Kalman filter are observers. We point out that if a 3D scheme such as a successive correction scheme is implemented repeatedly, it too can be expressed as an observer. This provides a way of looking at the dynamical properties of the resulting analysis. For example, we give conditions in the linear time invariant case under which the analysis will converge to the true solution. Using a simple model, we compare the results of data assimilation using the Cressman successive correction scheme and a robust observer.

Much of the thesis focuses on 4D variational assimilation methods, and in par-

ticular we address the problem of how to account for model error in these methods without incurring too much extra cost. Chapter 4 gives background on 4D variational methods of assimilation. We describe the strong constraint approach using the initial state as a control vector, and discuss the derivation of the adjoint models used in this approach. We then describe the correction term technique in which a constant correction term representing model error is added to the model equations, and used as a control vector as well as or instead of the initial state. Finally, we describe the weak constraint approach to variational assimilation, which allows for model error in a more general way, and refer to methods that have been proposed for solving this problem.

In Chapter 5, we concentrate on the correction term technique. We give conditions for uniqueness of the solution of the variational assimilation problem using the initial state, the correction term and both together as control vectors, and relate these conditions to the concept of *complete N-step observability*. We point out the importance of including a background estimate of the correction term in the cost function if there are insufficient observations; such a background term was not included in earlier published work on the correction term technique. We also compare the results of data assimilation using these different control vectors in a practical context, using a simple linear model with a constant source of model error. In the theory we present, we suggest that the correction term vector might have a dimension m less than the dimension n of the state vector. In these experiments, in which the source of model error is localised, this approach improves the efficiency of the method.

Then, in Chapter 6, we consider how we could use a more general representation of model error in variational assimilation. We give examples of different forms for representing model error supposing that model error is composed of serially correlated and serially uncorrelated components, and we discuss how the technique of *state augmentation* can be used to estimate the serially correlated component of model error along with the model state. We suggest a *generalized correction term technique* in which the correction term represents a serially correlated component of model error which might evolve in time and might have dimension m less than or

greater than the dimension of the model state. We carry out experiments using a simple model in which model error is not constant in time. In these experiments we use an "evolving correction term" which evolves as the model state does.

In Chapter 7, we carry out experiments using a 1D nonlinear shallow water model. We compare the results of data assimilation using the constant correction term, the initial state and both together as control vectors in the presence of different types of model error and errors in the initial state. In particular, we investigate whether the constant correction term can compensate for model error on a significant timescale, when model error depends on the model state. Finally, in Chapter 8, we summarize the conclusions from the work in the thesis and discuss how the work could be extended.

Throughout the thesis we bear in mind the application of data assimilation for numerical weather prediction in an operational context. Here, the huge dimension of the model state is a dominating factor in the practical choice of assimilation methods. Data assimilation is also used in other applications in the atmospheric and oceanic sciences, as we discussed in the previous section. Apart from these applications, state estimation using observed data has many applications in engineering, and the work in this thesis has relevance to this wider field also.

Chapter 2

Mathematical Background

Throughout this thesis, we will be looking at data assimilation for meteorology and oceanography using a framework of mathematical control theory. In the first section we introduce the general model system, using control theory notation. Then, in Section 2.2, we give background on some of the basic concepts of control theory which will be useful, and state definitions and theorems which will be referred to later on. In Section 2.3 we give background on nonlinear optimization theory, and in Section 2.4 describe descent algorithms that may be used to iterate to an optimal solution. Sections 2.3 and 2.4 provide the background for the *variational* data assimilation methods. Finally, Section 2.5 gives background on probability theory and the concept of a "most likely" estimate, which is widely used in data assimilation.

In this background chapter, we limit our discussion to *discrete* systems, since this is most convenient for application to numerical models of meteorology and oceanography. Many texts on control theory concentrate on continuous systems, with only brief reference to the discrete case. However, we treat the discrete case since the transition from the continuous to the discrete case is not always immediate [86].

2.1 Introducing the System

To start with, we introduce the general nonlinear model system which we will use throughout the thesis, and explain what we mean by the true model state and model

error. We specify how the observational data is related to the true model state. We then introduce the linear version of the system, which we will want to focus on in some situations.

2.1.1 The nonlinear model

We consider the discrete, nonlinear model on the time interval $[t_0, t_N]$,

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k), \qquad k = 0, .., N-1, \tag{2.1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ represents the *model state* at time t_k, $\mathbf{u}_k \in \mathbb{R}^m$ is a vector of *model inputs* at time t_k, and $\mathbf{f}_k : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a nonlinear function describing the *evolution* of the state from time t_k to time t_{k+1}. The state represents model variables defined on a spatial grid $\{r_j\}, j = 0, ...J$, which might represent one, two or three spatial dimensions. The model inputs in our context might include tunable model parameters, forcing terms or boundary conditions. Equation (2.1) represents an explicit, one-step model on a fixed spatial grid. We stick with this notation for simplicity, although the theory can usually be generalized to implicit or multi-timestep models.

We assume that specification of the model state $\mathbf{x}_j$ at time t_j and the inputs $\mathbf{u}_j, .., \mathbf{u}_{k-1}$ uniquely determines the model state $\mathbf{x}_k$ at time t_k, for any $k > j$. We also assume that $\mathbf{f}_k$ is differentiable with respect to $\mathbf{x}_k$ and $\mathbf{u}_k$ for all k.

2.1.2 The true model state and model error

The development here, which introduces the concept of the *true model state* and defines *model error*, follows that in the papers by Cohn and Dee [17], Dee [25] and Cohn [16].

We suppose that the *true state* of the atmosphere or ocean at any time t in a time interval $[t_0, t_N]$ can be represented by a vector $\boldsymbol{\xi}(t)$ belonging to an infinite dimensional space $\mathcal{U}$. We further suppose that the evolution of the state from time t_k to time t_{k+1} can be described by a well-posed nonlinear system of equations, and can be written in the form

$$\boldsymbol{\xi}(t_{k+1}) = \boldsymbol{\psi}_k(\boldsymbol{\xi}(t_k)), \tag{2.2}$$

where $\boldsymbol{\psi}_k : \mathcal{U} \to \mathcal{U}$ is a uniquely defined nonlinear solution operator [17]. We now define the *true model state* $\mathbf{x}_k^t$ to be the representation of the infinite dimensional true state at time t_k on the model grid, and we write

$$\mathbf{x}_k^t = \Pi \boldsymbol{\xi}(t_k), \tag{2.3}$$

where $\Pi : \mathcal{U} \to \mathbb{R}^n$ is a mapping onto the model grid. Hence we can write the evolution of the true model state in terms of our model (2.1) as follows

$$\mathbf{x}_{k+1}^t = \mathbf{f}_k(\mathbf{x}_k^t, \mathbf{u}_k) + \boldsymbol{\varepsilon}_k, \qquad k = 0, .., N-1, \tag{2.4}$$

where

$$\boldsymbol{\varepsilon}_k = \Pi \boldsymbol{\psi}_k(\boldsymbol{\xi}(t_k)) - \mathbf{f}_k(\mathbf{x}_k^t, \mathbf{u}_k). \tag{2.5}$$

The term $\boldsymbol{\varepsilon}_k \in \mathbb{R}^n$ is the *model error* in the evolution operator $\mathbf{f}_k$. If, for example, the equation representing the evolution of the true state $\boldsymbol{\xi}(t)$ is a known system of partial differential equations, and if the model (2.1) is a *consistent* discretization of this, then model error is just the *truncation error* of the discretization. In general, however, as well as errors due to lack of resolution, sources of model error arise due to lack of knowledge of the true evolution of the atmosphere or oceans, or due to deliberate simplification of their known evolution. Hence, sources of model error include misspecification of model parameters, forcing terms and boundary conditions. Equation (2.5) shows that in general model error depends on the unknown true state in an unknown way [25], and so might be treated as stochastic forcing [16], or by some simple deterministic correction [26]. In Chapter 6 we consider specific examples of how we might approximate or represent the model error term so we can account for it in data assimilation.

2.1.3 Observational data

We suppose we have a set of observations $\mathbf{y}_0, .., \mathbf{y}_{N-1}$ which are related to the true model state by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1, \tag{2.6}$$

where $\mathbf{y}_k \in \mathbb{R}^{p_k}$ is a vector of p_k observations at time t_k, $\mathbf{h}_k : \mathbb{R}^n \to \mathbb{R}^{p_k}$ is a nonlinear function relating the observations to the model state at time t_k, and $\boldsymbol{\delta}_k \in$

$\mathbb{R}^{p_k}$ represents the *observational error* at time t_k. In the context of control theory, the observations are generally referred to as model *outputs*. If observation times do not coincide with the timesteps t_k, then $\mathbf{h}_k$ will include temporal interpolation. The number of observations p_k varies with time, and this includes the possibility of no observations at some timesteps.

Observational error has two components usually referred to as *measurement errors* and *representational errors*. The measurement errors are due to errors in the measurement instruments and in the transmission of information, and the representation errors are due to errors in $\mathbf{h}_k$. More detail on the form of observational error is given in [24].

2.1.4 The linear assimilation system

In some cases we will limit our attention to linear theory, and so we consider the discrete, linear, time-varying model

$$\mathbf{x}_{k+1} = A_k \mathbf{x}_k + B_k \mathbf{u}_k, \tag{2.7}$$

with $\mathbf{x}_k$ and $\mathbf{u}_k$ defined as in (2.1), and with $A_k \in \mathbb{R}^{n \times n}$, $B_k \in \mathbb{R}^{n \times m}$. We assume that A_k is nonsingular and that B_k has rank m for all k, so that specification of $\mathbf{x}_0$ and the $\mathbf{u}_j$, $j = 0, .., k-1$ uniquely determines $\mathbf{x}_k$ for $k > 0$. We suppose that the evolution of the true model state $\mathbf{x}_k^t$ satisfies

$$\mathbf{x}_{k+1}^t = A_k \mathbf{x}_k^t + B_k \mathbf{u}_k + \boldsymbol{\varepsilon}_k, \qquad k = 0, .., N-1, \tag{2.8}$$

where $\boldsymbol{\varepsilon}_k \in \mathbb{R}^n$ is the model error as defined in Subsection 2.1.2.

We now suppose that the observations are related linearly to the true model state as follows,

$$\mathbf{y}_k = C_k \mathbf{x}_k^t + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1, \tag{2.9}$$

with $\mathbf{y}_k$ and $\boldsymbol{\delta}_k$ defined as in (2.6) and $C_k \in \mathbb{R}^{p_k \times n}$.

If the assimilation system (2.8),(2.9) is a linearization of the system (2.4),(2.6) about some reference state $\mathbf{x}_k^o$ and input $\mathbf{u}_k^o$, then A_k and C_k are the Jacobians of $\mathbf{f}_k$ and $\mathbf{h}_k$ respectively with respect to $\mathbf{x}_k$, and B_k is the Jacobian of $\mathbf{f}_k$ with respect to $\mathbf{u}_k$, all evaluated at $(\mathbf{x}_k^o, \mathbf{u}_k^o)$. In this case, the model (2.8) is often referred to as the *tangent linear model* of (2.4) in data assimilation literature [48], [20].

2.1.5 State transition matrix

For some applications of the linear system, it will be useful to relate the state at a given time to the state at any earlier time. We therefore introduce the *state transition matrix* $\Phi(k, j)$, for the unforced system

$$\mathbf{x}_{k+1} = A_k \mathbf{x}_k, \tag{2.10}$$

which relates the state at time t_k to the state at an earlier time t_j as follows, [2],

$$\mathbf{x}_k = \Phi(k, j)\mathbf{x}_j \qquad \forall k \geq j, \tag{2.11}$$

with

$$\Phi(j, j) = I \qquad \forall j. \tag{2.12}$$

For the system (2.10) the state transition matrix is given uniquely by

$$\Phi(k, j) = \prod_{i=j}^{k-1} A_i. \tag{2.13}$$

Clearly we have

$$\Phi(l, j) = \Phi(l, k)\Phi(k, j) \qquad \forall l \geq k \geq j, \tag{2.14}$$

and since the matrices A_i are assumed to be nonsingular, we also may define

$$\Phi(j, k) = \Phi^{-1}(k, j), \qquad \forall j \leq k. \tag{2.15}$$

For the forced model (2.7), we now have [2]

$$\mathbf{x}_k = \Phi(k, j)\mathbf{x}_j + \sum_{i=j}^{k-1} \Phi(k, i+1)B_i \mathbf{u}_i \qquad \forall k > j. \tag{2.16}$$

The relationship (2.16) will be important later on in the thesis.

2.2 Controllability and Observability

The general aim of control theory is to regulate the state to some desired state by a suitable choice of the inputs which we are free to choose. The variables we use to manipulate the state are known as *control variables*. Generally, the model inputs are used as control variables. In some cases we might be free to choose the initial

states, and so these too could be used as control variables. The "strong constraint" approach to variational assimilation hinges on the use of the initial state as a *control vector*, or vector of control variables, since the idea is to choose that initial state which will ensure that the state at later times is as desired. One of the areas we will investigate is the use of correction terms representing model error as control vectors. Other work has been carried out using tunable model parameters [74] or boundary conditions [50] as control variables.

In this section, we introduce the concepts of controllability and observability, and give some theoretical results which can be used to determine whether a system is controllable or observable.

2.2.1 Some definitions

The concepts of controllability and observability are very important in control theory. The concept of *controllability* addresses the question of whether it is possible to choose control variables to obtain the desired state, and the concept of *observability* addresses whether it is possible to reconstruct the model state from the outputs or observations and a knowledge of the model inputs. Here we give definitions for complete μ-step controllability and complete ν-step observability. Often, the phrase "μ-step" or "ν-step" is not included in definitions of controllability or observability. In the theory we present in Chapter 5, however, we require these more specific μ- and ν-step definitions.

The definitions are for the linear system with no model error and no observational error, ie for the system

$$\mathbf{x}_{k+1}^t = A_k \mathbf{x}_k^t + B_k \mathbf{u}_k, \tag{2.17}$$

$$\mathbf{y}_k = C_k \mathbf{x}_k^t. \tag{2.18}$$

However, as we will see in Chapter 5, the concepts are still useful for the system (2.8),(2.9) with model error and observational error.

We note that since the system matrices A_k are nonsingular, the related concepts of *reachability* and *detectability* are in this case equivalent to controllability and observability respectively, [86].

Definition 2.1 The system (2.17),(2.18) is *completely μ-step controllable at time t_j* if for any arbitrary state $\mathbf{x}_j$ at time t_j and any desired state $\mathbf{x}^d$, there is an admissible control sequence $\mathbf{u}_j, .., \mathbf{u}_{j+\mu-1}$ on the discrete time interval $[t_j, t_{j+\mu-1}]$ which drives the system to the desired state $\mathbf{x}^d$ at time $t_{j+\mu}$.

If the system is completely μ-step controllable for any time t_j, it is *completely μ-step controllable.*

If the system is completely μ-step controllable (at time t_j) for some μ, we might simply say that the system is *completely controllable (at time t_j).*

Definition 2.2 The system (2.17),(2.18) is *completely ν-step observable at time t_j* if and only if knowledge of the outputs $\mathbf{y}_j, \mathbf{y}_{j+1}, .., \mathbf{y}_{j+\nu-1}$ and of the inputs $\mathbf{u}_j, \mathbf{u}_{j+1}, .., \mathbf{u}_{j+\nu-2}$ is sufficient to determine the state $\mathbf{x}_j$.

If the system is completely ν-step observable for any time t_j, it is *completely ν-step observable.*

If the system is completely ν-step observable (at time t_j) for some ν, we might simply say that the system is *completely observable (at time t_j).*

2.2.2 Theory for the general linear case

For the linear system (2.17),(2.18), the following theorems can be used to determine whether the system is controllable or observable. We first introduce the *μ-step controllability matrix $\mathcal{C}_\mu^j$ for time t_j* and the *ν-step observability matrix $\mathcal{O}_\nu^j$, for time t_j* as follows.

$$\mathcal{C}_\mu^j = (B_{j-1}, \Phi(j, j-1)B_{j-2}, \ldots, \Phi(j, j-\mu+1)B_{j-\mu}), \tag{2.19}$$

$$\mathcal{O}_\nu^j = \begin{pmatrix} C_j \\ C_{j+1}\Phi(j+1, j) \\ \vdots \\ C_{j+\nu-1}\Phi(j+\nu-1, j) \end{pmatrix}. \tag{2.20}$$

Theorem 2.1 *The linear system (2.17),(2.18) is completely μ-step controllable at time t_j if and only if Rank $(\mathcal{C}_\mu^j) = n$.*

The proof of Theorem 1 is given in [86] for the concept of reachability, which in this case is equivalent to that of controllability. We note that complete μ-step controllability at time t_j implies complete μ'-step controllability at time t_j for all integers $\mu' \geq \mu$ [86].

We give the proof of the next theorem, since it illustrates a line of argument we will use later. The proof is based on that given by Weiss [86], but uses our notation and expresses a couple of the arguments slightly differently.

Theorem 2.2 *The linear system (2.17),(2.18) is completely ν-step observable at time t_j if and only if Rank $(\mathcal{O}_\nu^j) = n$.*

Proof

(i) We firstly show that Rank $(\mathcal{O}_\nu^j) = n$ is a sufficient condition for complete ν-step observability at time t_j. We suppose that Rank $(\mathcal{O}_\nu^j) = n$. We use (2.16) and (2.18) to rewrite the information available from the observations on the time interval $[t_j, t_{j+\nu-1}]$ explicitly in terms of the state $\mathbf{x}_j$, as follows

$$\mathbf{y}_k = C_k \Phi(k, j)\mathbf{x}_j + \mathbf{b}_k, \qquad k = j, ..., j+\nu-1, \tag{2.21}$$

where $\mathbf{b}_j = 0$ and

$$\mathbf{b}_k = C_k \sum_{i=j}^{k-1} \Phi(k, i+1) B_i \mathbf{u}_i \qquad k = j+1, ..., j+\nu-1. \tag{2.22}$$

Hence we can write

$$\mathcal{O}_\nu^j \mathbf{x}_j = \mathbf{z}, \tag{2.23}$$

where

$$\mathbf{z} = \begin{pmatrix} \mathbf{y}_j - \mathbf{b}_j \\ \mathbf{y}_{j+1} - \mathbf{b}_{j+1} \\ \vdots \\ \mathbf{y}_{j+\nu-1} - \mathbf{b}_{j+\nu-1} \end{pmatrix}. \tag{2.24}$$

Since Rank $(\mathcal{O}_\nu^j) = n$ and, by construction, $\mathbf{z}$ is a linear combination of the columns of $\mathcal{O}_\nu^j$ so that Rank $(\mathcal{O}_\nu^j | \mathbf{z}) = n$, $\mathbf{x}_j$ can be uniquely determined from the observations and specified inputs.

(ii) To show that Rank $(\mathcal{O}_\nu^j) = n$ is a necessary condition for complete ν-step observability at time t_j, we suppose that the system is completely ν-step observable at time t_j, but that Rank $(\mathcal{O}_\nu^j) < n$, and let $\mathbf{u}_k = 0$, $k = j, .., j+\nu-2$.
Then there exists a *nonzero* vector $\mathbf{v} \in \mathbb{R}^n$, such that

$$\mathcal{O}_\nu^j \mathbf{v} = 0. \tag{2.25}$$

Putting $\mathbf{x}_j = \mathbf{v}$ in (2.23) with zero input, we have $\mathbf{z} = 0$, which violates complete μ-step observability (since we have zero output over the whole time interval $[t_j, t_{j+\nu-1}]$ although the state at time t_j is not zero). □

We note that complete ν-step observability at time t_j implies complete ν'-step observability at time t_j for any integer $\nu' \geq \nu$ [86].

2.2.3 Theory for the linear, time-invariant case

The results given above can be applied to the time-invariant system, but in this special case, we can say a bit more. The linear, time-invariant system is given by

$$\mathbf{x}_{k+1}^t = A\mathbf{x}_k^t + B\mathbf{u}_k, \tag{2.26}$$

$$\mathbf{y}_k = C\mathbf{x}_k^t, \tag{2.27}$$

where $\mathbf{x}_k^t \in \mathbb{R}^n$, $\mathbf{u}_k \in \mathbb{R}^m$ and $\mathbf{y}_k \in \mathbb{R}^p$ are defined as in (2.4) and (2.6), and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and $C \in \mathbb{R}^{p\times n}$ are constant matrices. For a time-invariant system, μ-step controllability at time t_j clearly implies μ-step controllability for all time, and ν-step observability at time t_j implies ν-step observability for all time. We introduce the notation

$$\mathcal{C}_\mu^* = (B, AB, ..., A^{\mu-1}B), \qquad \mathcal{O}_\nu^* = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{\nu-1} \end{pmatrix}. \tag{2.28}$$

The time invariant system (2.26),(2.27) is completely controllable if and only if Rank $(\mathcal{C}_n^*) = n$, [2], and is completely observable if and only if Rank $(\mathcal{O}_n^*) = n$, [66].

Later in the thesis, we will want to apply theoretical results involving the concept of complete ν-step observability to the special case of a time invariant system, and where it is possible, to express the results in terms of the more familiar concept of complete observability. The following theorem enables us to do this.

Theorem 2.3 *a) If the linear time-invariant system is not completely observable, then it is not completely ν-step observable for all positive integers ν.*
b) If $\nu \geq n$ then the linear time-invariant system is completely ν-step observable if and only if it is completely observable.

Proof

a) We must show that $\text{Rank}(\mathcal{O}_n^*) < n$ implies $\text{Rank}(\mathcal{O}_\nu^*) < n$ for all positive integers ν, and we do this by showing

$$\text{Rank}(\mathcal{O}_\nu^*) \leq \text{Rank}(\mathcal{O}_n^*) \tag{2.29}$$

for all ν.
This is clearly true for $\nu \leq n$. We suppose that $\nu = n + 1$. By the Cayley Hamilton theorem [2], we have

$$A^n = \sum_{j=0}^{n-1} \gamma_j A^j \tag{2.30}$$

for some $\gamma_j \in \mathbb{R}$, and so CA^n can be be written as a linear combination of the rows of $\mathcal{O}_n^*$, and hence (2.29) holds. Similarly, for any $\nu > n$,

$$A^\nu = (\sum_{j=0}^{n-1} \gamma_j A^j) A^{\nu-n}, \tag{2.31}$$

and hence CA^ν is still a linear combination of the rows of $\mathcal{O}_n^*$, and so (2.29) holds for all positive integers ν.

b) It follows from part a) that for any positive integer ν, the linear time-invariant system is completely ν-step observable only if it is completely observable. We now suppose that the linear time invariant system is completely observable, and hence is completely n-step observable. As noted earlier, complete n-step observability

implies complete ν-step observability for any $\nu \geq n$, and so part b) of the theorem holds. □

One further result which will be useful when considering the time invariant case is the Hautus condition [28], which is given in Theorem 2.4.

Theorem 2.4 *The linear time-invariant system (2.26),(2.27) is completely observable if and only if,* $\forall\ \lambda \in \mathbb{C}$ *and* $\forall\ \mathbf{s} \in I\!R^n$,
$(A - \lambda I)\mathbf{s} = 0$ *and* $C\mathbf{s} = 0 \Leftrightarrow \mathbf{s} = 0$.

2.3 Nonlinear optimization theory

The theory we give here provides background for the variational methods of data assimilation which we investigate in this thesis. Useful texts for this material include [36], [29], [43], [10], and [89].

2.3.1 Preliminaries

A *Hilbert space* is a complete, linear inner-product space. All the properties of Hilbert spaces are important for our purposes [43]. We denote the inner product defined on a Hilbert space $\mathcal{V}$ by $< \mathbf{x}, \mathbf{y} >_{\mathcal{V}}$, for any two elements $\mathbf{x}$ and $\mathbf{y} \in \mathcal{V}$. We note that real, n-dimensional Euclidean space $\mathbb{R}^n$ with the Euclidean inner product (or "dot product") is a Hilbert space, and throughout the thesis use the notation

$$< \mathbf{x}, \mathbf{y} >:= \mathbf{x}^T\mathbf{y} \tag{2.32}$$

to refer to this inner product.

Later in the thesis, we refer to the *adjoint* of a linear operator. For a linear operator A from a Hilbert space $\mathcal{U}$ to a Hilbert space $\mathcal{V}$, the *adjoint operator* A^* is the linear operator from $\mathcal{V}$ to $\mathcal{U}$ for which, for all $\mathbf{u} \in \mathcal{U}$ and $\mathbf{v} \in \mathcal{V}$

$$< \mathbf{v}, A\mathbf{u} >_{\mathcal{V}} = < A^*\mathbf{v}, \mathbf{u} >_{\mathcal{U}} . \tag{2.33}$$

In the case where $\mathcal{U}$ is $\mathbb{R}^m$ and $\mathcal{V}$ is $\mathbb{R}^n$, both with the Euclidean product (or dot product), $A : \mathbb{R}^m \rightarrow \mathbb{R}^n$ is an $n \times m$ matrix and we have

$$< \mathbf{v}, A\mathbf{u} >= \mathbf{v}^T A\mathbf{u} = (A^T\mathbf{v})^T\mathbf{u} = < A^T\mathbf{v}, \mathbf{u} >, \tag{2.34}$$

so that $A^T : \mathbb{R}^n \to \mathbb{R}^m$ is the adjoint of A.

We consider a nonlinear, real valued function $\mathcal{J}$ on $\mathcal{V}$. We suppose that $\mathcal{J}$ is three times differentiable at $\mathbf{v}_0 \in \mathcal{V}$, and that $\mathbf{v}_0 + \theta\boldsymbol{\delta}\mathbf{v} \in \mathcal{V}$ represents a perturbation of size $\theta \in [-1, 1]$ in a direction $\boldsymbol{\delta}\mathbf{v}$ from $\mathbf{v}_0$. The Taylor series expansion of $\mathcal{J}$ about $\mathbf{v}_0$ can be written as follows [43]

$$\mathcal{J}(\mathbf{v}_0 + \theta\boldsymbol{\delta}\mathbf{v}) = \mathcal{J}(\mathbf{v}_0) + \theta < \nabla_{\mathbf{v}}\mathcal{J}(\mathbf{v}_0), \boldsymbol{\delta}\mathbf{v} >_{\mathcal{V}} + \theta^2 < \boldsymbol{\delta}\mathbf{v}, \mathcal{H}_{\mathbf{v}}(\mathbf{v}_0)\boldsymbol{\delta}\mathbf{v} >_{\mathcal{V}} + O^3(\theta), \tag{2.35}$$

where the vector $\nabla_{\mathbf{v}}\mathcal{J}(\mathbf{v}_0) \in \mathcal{V}$ is the *gradient* of $\mathcal{J}$ with respect to $\mathbf{v}$ at $\mathbf{v}_0$, and the linear operator $\mathcal{H}_{\mathbf{v}}(\mathbf{v}_0) : \mathcal{V} \to \mathcal{V}$ is the *Hessian* of $\mathcal{J}$ with respect to $\mathbf{v}$ at $\mathbf{v}_0$. Throughout the thesis, we use this notation to denote the gradient and the Hessian of a real valued function.

2.3.2 Unconstrained minimization

We suppose that we wish to minimize a real valued function $\mathcal{J}$, usually referred to as a *cost function*, which is defined on a Hilbert space $\mathcal{V}$. The unconstrained minimization problem we consider is

Problem $\mathcal{U}$:

Minimize $\mathcal{J}$; ie, find $\mathbf{v}^* \in \mathcal{V}$ such that

$$\mathcal{J}(\mathbf{v}^*) \leq \mathcal{J}(\mathbf{v}) \tag{2.36}$$

for all $\mathbf{v}$ in some neighbourhood $\mathcal{N} \subseteq \mathcal{V}$ of $\mathbf{v}^*$.

If such a $\mathbf{v}^*$ exists, it is called a *local minimum* of $\mathcal{J}$. If the inequality in (2.36) is strict, then $\mathbf{v}^*$ is a *unique* local minimum. If $\mathcal{N}(\mathbf{v}^*) = \mathcal{V}$, then $\mathbf{v}^*$ is also a *global* minimum.

Since we have no constraints, the following is a necessary condition for $\mathbf{v}^*$ to minimize $\mathcal{J}$

$$\nabla_{\mathbf{v}}\mathcal{J}(\mathbf{v}^*) = 0. \tag{2.37}$$

In the special case that the cost function is quadratic in $\mathbf{v}$,

$$\mathcal{J} = \frac{1}{2} < \mathbf{v}, A\mathbf{v} >_{\mathcal{V}} + < \mathbf{b}, \mathbf{v} >_{\mathcal{V}} + c, \tag{2.38}$$

where $\mathbf{b} \in \mathcal{V}$ and $c \in \mathbb{R}$ are constants and $A : \mathcal{V} \to \mathcal{V}$ is a linear operator, if A is a positive definite operator, then a minimum $\mathbf{v}^*$ exists, is unique, and is given by $\mathbf{v}^* = -A^{-1}\mathbf{b}$, [43]. If, however, A is only positive semi-definite, a minimum $\mathbf{v}^*$ exists but is not unique, since $\mathbf{v}^* + \mathbf{z}$ is also a minimum for any $\mathbf{z}$ satisfying $< \mathbf{z}, A\mathbf{z} >= 0$. Further, if A is indefinite, then there is no minimum.

We now return to the general case where $\mathcal{J}$ is not necessarily linear or quadratic. We suppose $\mathcal{J}$ is three times differentiable, and so can be expanded in a Taylor series of the form (2.35). Then, for $\|\theta\boldsymbol{\delta}\mathbf{v}\|$ small enough, the quadratic part of the expansion dominates, so if $\nabla_{\mathbf{v}}(\mathbf{v}^*) = 0$, and $\mathcal{H}_{\mathbf{v}}(\mathbf{v}^*)$ is a positive definite operator, then $\mathbf{v}^*$ is a unique local minimum of $\mathcal{J}$ [43]. If $\mathcal{H}_{\mathbf{v}}(\mathbf{v}^*)$ is only a positive semi-definite operator, we can draw no conclusions about $\mathbf{v}^*$, because of the influence of the higher order terms in the expansion. However, if $\mathcal{H}_{\mathbf{v}}(\mathbf{v}^*)$ is indefinite, then $\mathbf{v}^*$ cannot be a minimum.

2.3.3 Constrained minimization

In this subsection, we consider constrained minimization of a real valued function $\mathcal{J}$ over $\mathbb{R}^n$, which with the Euclidean inner product (2.32) is a Hilbert space.

The constrained minimization problem we consider is

Problem $\mathcal{C}$:

Minimize $\mathcal{J}$ subject to the r constraints

$$g_k(\mathbf{v}) = 0, \qquad k = 1, .., r, \tag{2.39}$$

or equivalently

$$\mathbf{g}(\mathbf{v}) = 0, \tag{2.40}$$

where $r \leq n$ and $\mathbf{g}$ is a vector of r real valued functions $g_k : \mathbb{R}^n \to \mathbb{R}$, $k = 1, .., r$ which are continuously differentiable. We further assume that the vectors $\nabla_{\mathbf{v}} g_k(\mathbf{v})$, $k = 1, .., r$ are linearly independent for all $\mathbf{v} \in \mathbb{R}^n$.

A constrained minimization problem of this form can be addressed as an unconstrained optimization problem using the technique of Lagrange multipliers. The

Lagrangian function associated with Problem $\mathcal{C}$ is defined to be

$$\mathcal{L}(\mathbf{v}, \boldsymbol{\lambda}) = \mathcal{J}(\mathbf{v}) + \boldsymbol{\lambda}^T \mathbf{g}(\mathbf{v}), \tag{2.41}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^r$ is a vector of r Lagrange multipliers λ_k. A solution of Problem $\mathcal{C}$, if it exists, can be found by extremizing the (unconstrained) Lagrangian function $\mathcal{L}$ with respect to $\mathbf{v}$ and $\boldsymbol{\lambda}$. Necessary conditions for an extremal are [29],

$$\nabla_{\mathbf{v}} \mathcal{L} = 0, \tag{2.42}$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L} = 0. \tag{2.43}$$

Any vector $\mathbf{v} \in \mathbb{R}^n$ satisfying (2.40) can be written in the form

$$\mathbf{v} = \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix} \tag{2.44}$$

with $\mathbf{u} \in \mathbb{R}^{n-r}$ and $\mathbf{x} \in \mathbb{R}^r$, where the $n-r$ components u_j may be chosen independently, and the r components are determined from the choice of the u_j through (2.40) [89]. We refer to the $n-r$ variables u_j as *control variables*, and the vector $\mathbf{u}$ as a *control vector*.

2.3.4 Solving Problem $\mathcal{C}$ by reducing the control vector

We now describe an iterative method for finding $\mathbf{v}^*$ satisfying necessary conditions for a solution of Problem $\mathcal{C}$ by iterating on the control variables. Since this involves iterating on the control vector $\mathbf{u}$ rather than on the full vector $\mathbf{v}$, this technique is referred to as "reduction of the control vector". This method was suggested for application to 4D variational assimilation by Le Dimet and Talagrand [51], who used the optimal control approach of Lions [53] rather than the Lagrange multiplier approach we use here.

Necessary conditions for an extremal of $\mathcal{L}$ are given by

$$\nabla_{\mathbf{u}} \mathcal{L} = \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{v}) + G_{\mathbf{u}}^T(\mathbf{v}) \boldsymbol{\lambda} = 0, \tag{2.45}$$

$$\nabla_{\mathbf{x}} \mathcal{L} = \nabla_{\mathbf{x}} \mathcal{J}(\mathbf{v}) + G_{\mathbf{x}}^T(\mathbf{v}) \boldsymbol{\lambda} = 0, \tag{2.46}$$

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L} = \mathbf{g}(\mathbf{v}) = 0, \tag{2.47}$$

where $G_{\mathbf{u}} \in \mathbb{R}^{r\times(n-r)}$ and $G_{\mathbf{x}} \in \mathbb{R}^{r\times r}$ are the Jacobian matrices of $\mathbf{g}$ with respect to $\mathbf{u}$ and $\mathbf{x}$ respectively. Since the vectors $\nabla_{\mathbf{v}} g_k(\mathbf{v})$ for $k = 1,..,r$ are linearly independent, the Jacobian $G_{\mathbf{x}}(\mathbf{v})$ is invertible.

From a guess $\mathbf{u}$ for the control vector, the corresponding vector $\mathbf{x}$ is specified from the constraints (2.40), and hence (2.47) holds. From this choice of $\mathbf{v} = \begin{pmatrix} \mathbf{u} \\ \mathbf{x} \end{pmatrix}$, $\boldsymbol{\lambda}$ can be uniquely chosen to satisfy (2.46). Then we have the following expression for the gradient of $\mathcal{L}$ with respect to the control vector $\mathbf{u}$

$$\nabla_{\mathbf{u}}\mathcal{L} = \nabla_{\mathbf{u}}\mathcal{J}(\mathbf{v}) + G_{\mathbf{u}}^T(\mathbf{v})\boldsymbol{\lambda}. \tag{2.48}$$

This gradient can be used in a gradient method to obtain a better guess of $\mathbf{u}$, and the procedure repeated until (2.45) holds.

2.4 Gradient methods

We consider here the problem of unconstrained minimization of a cost function $\mathcal{J}$ over $\mathbb{R}^n$ with respect to a control vector $\mathbf{u} \in \mathbb{R}^n$. We suppose that for any guess $\mathbf{u}^k$ of an optimal $\mathbf{u}$, we can find $\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k)$, the gradient of the function with respect to the control vector at $\mathbf{u}_k$.

A *gradient method* for iterating to a minimizing $\mathbf{u}^*$ is of the following general form [81],

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \rho^k G_k \mathbf{d}^k \tag{2.49}$$

where $\mathbf{d}^k \in \mathbb{R}^n$ is the descent direction based on the gradient $\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k)$, $\rho^k \in \mathbb{R}$ is the step-length, and $G_k \in \mathbb{R}^{n\times n}$ is a matrix which should ideally approximate the inverse of the Hessian $\mathcal{H}_{\mathbf{u}}(\mathbf{u}^k)$ of $\mathcal{J}$ with respect to $\mathbf{u}$ at $\mathbf{u}^k$.

We now outline three types of gradient algorithms; steepest descent methods, conjugate gradient methods and Newton-type methods. We give more detail on the conjugate gradient method and a package quasi-Newton method, since we use these methods in the thesis.

2.4.1 The steepest descent algorithm

In this case, the direction $\mathbf{d}^k$ in (2.49) is simply the direction $\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k)$, G_k is the identity, and ρ^k is chosen to ensure

$$\mathcal{J}(\mathbf{u}^{k+1}) < \mathcal{J}(\mathbf{u}^k). \tag{2.50}$$

In practice, this might be done by setting $\rho^k = 1$ initially on each iteration, and halving ρ^k until (2.50) holds. Alternative step-length choices are given in [36].

The advantage of the steepest descent method lies in its simplicity, but the rather ad-hoc method of finding the step-length can render it very inefficient since it involves many evaluations of $\mathcal{J}$ and $\nabla_{\mathbf{u}}\mathcal{J}$. Further, choosing the direction $\mathbf{d}^k$ with no consideration of the previous directions used is not the most efficient approach. The conjugate gradient method provides a more sophisticated approach to calculating ρ^k and $\mathbf{d}^k$, and we describe this next.

2.4.2 The conjugate gradient method

The aim of the conjugate gradient method (CGM) is to choose the k^{th} descent direction $\mathbf{d}^k$ to be a projection of the gradient $\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k)$ onto a subspace of $\mathbb{R}^n$ which is orthogonal to $\mathbf{d}^j$ for $j = 0, 1, .., k-1$. Primarily, the CGM addresses an unconstrained minimization problem with quadratic cost function,

$$\mathcal{J} = \frac{1}{2} < \mathbf{u}, A\mathbf{u} > + < \mathbf{b}, \mathbf{u} >, \tag{2.51}$$

where $A \in \mathbb{R}^{n\times n}$ is symmetric, positive definite, and $\mathbf{b} \in \mathbb{R}^n$. The method calculates the optimal step-length ρ^k for each direction, and so for the quadratic case above should theoretically converge in at most n iterations. However, this condition does not hold in practice because of rounding errors, and if n is large, we require good convergence in far fewer iterations in any case.

The conjugate gradient iteration on $\mathbf{u}$ is given by

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \rho^k \mathbf{d}^k \tag{2.52}$$

$$\mathbf{d}^{k+1} = -\mathbf{r}^{k+1} + \beta^k \mathbf{d}^k, \tag{2.53}$$

where

$$\rho^k = \frac{< \mathbf{r}^k, \mathbf{d}^k >}{< \mathbf{d}^k, A\mathbf{d}^k >}, \qquad \beta^k = \frac{< \mathbf{r}^{k+1}, A\mathbf{d}^k >}{< \mathbf{d}^k, A\mathbf{d}^k >}, \tag{2.54}$$

and

$$\mathbf{r}^k = A\mathbf{u}^k + \mathbf{b} = \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^k) \tag{2.55}$$

with

$$\mathbf{d}^0 = -\mathbf{r}^0. \tag{2.56}$$

The conjugate gradient method can also be used when $\mathcal{J}$ is not quadratic, as described in [43]. For the non-quadratic case, however, the step-length ρ^k given by (2.54) is no longer the "exact" step-length for the direction $\mathbf{d}^k$, and a different procedure (a linesearch) must be used to give a good estimate of the optimal step-length. This need for an accurate step-length can lead to expensive line searches for non-quadratic problems. However, Newton-type methods have the advantage that accurate line searches for the optimal step-length are not needed.

2.4.3 Newton's method and quasi-Newton methods

Newton's method

Newton's method provides an iterative solution to the problem

$$\mathbf{f}(\mathbf{u}) = 0, \tag{2.57}$$

where $\mathbf{u} \in \mathbb{R}^n$, and $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear function, which is assumed to be continuously differentiable in the neighbourhood of $\mathbf{u}$ with nonsingular Jacobian $F_{\mathbf{u}}(\mathbf{u})$. Newton's method for solving (2.57) is

$$\mathbf{u}^{k+1} = \mathbf{u}^k - F_{\mathbf{u}}^{-1}(\mathbf{u}^k)\mathbf{f}(\mathbf{u}^k). \tag{2.58}$$

Newton's method has fast convergence (quadratic rate), and for a quadratic cost function converges in just one iteration.

In the context of our minimization problem, the problem of the form (2.57) that we wish to solve is

$$\nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}) = 0. \tag{2.59}$$

For this problem, Newton's method is

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \mathcal{H}_{\mathbf{u}}^{-1}(\mathbf{u}^k)\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k). \tag{2.60}$$

The drawback of Newton's method however is that it requires a solution of the equation

$$\mathcal{H}_{\mathbf{u}}(\mathbf{u}^k)(\mathbf{u}^k - \mathbf{u}^{k+1}) = \nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k) \tag{2.61}$$

at each iteration. For this reason, modifications of the Newton method have been devised, to simplify the Hessian or to approximate its inverse. These modifications constitute the *quasi-Newton* methods.

Quasi-Newton methods

Quasi-Newton methods for our minimization problem are of the form

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \rho^k G_k \nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k) \tag{2.62}$$

where G_k approximates $\mathcal{H}_{\mathbf{u}}^{-1}(\mathbf{u}^k)$, the inverse Hessian at $\mathbf{u}^k$. A particular class of quasi-Newton methods is the class of methods which use information on the previous gradients to compose G_k, and so to gradually build up a better approximation of the true inverse Hessian. The BFGS update formula for G_k [81] has been widely considered one of the most efficient [65], [81], [33]. For problems where n is large, however (say, $n > 500$, [33]), the cost of storing these approximate Hessian matrices becomes prohibitively expensive, with a memory requirement of $O(n^2)$, compared to the $O(n)$ memory requirement of the CGM.

This problem may be alleviated by storing only the most recent gradient information, from, say, the last $\hat{m}$ iterations [65]; such methods are called *limited memory* quasi-Newton methods.

Another important issue for quasi-Newton methods is the condition number of the matrices G_k. Large condition numbers lead to large round-off errors, which affect the numerical stability of the method. This matter is treated by Oren and Spedicato [67]. We now give some detail on a limited-memory quasi-Newton algorithm used in a package from INRIA. We use the program N1QN3.f in the work described in Chapter 7.

The INRIA N1QN3 minimization algorithm

This minimization algorithm uses the the quasi-Newton update formula (2.62), in which G_k, the current approximation of the inverse Hessian, is calculated using a limited memory BFGS update. It is based on an algorithm by Nocedal, [65], with an added preconditioning option, and is described in the documentation [34] and in the paper by Gilbert and Lamaréchal [33].

The general *inverse BFGS formula*, for approximating the new inverse Hessian G_{k+1} from G_k is as follows

$$G_{k+1} = (I - \frac{\mathbf{s}^k(\mathbf{y}^k)^T}{(\mathbf{y}^k)^T\mathbf{s}^k})G_k(I - \frac{\mathbf{y}^k(\mathbf{s}^k)^T}{(\mathbf{y}^k)^T\mathbf{s}^k} + \frac{\mathbf{s}^k(\mathbf{s}^k)^T}{(\mathbf{y}^k)^T\mathbf{s}^k}), \tag{2.63}$$

where

$$\mathbf{s}^k = \mathbf{u}^{k+1} - \mathbf{u}^k, \qquad \mathbf{y}^k = \nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^{k+1}) - \nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}^k). \tag{2.64}$$

The matrix G_k is not stored explicitly in memory, but the product $G_k\nabla_{\mathbf{u}}\mathcal{J}(\mathbf{u}_k)$ is calculated from a diagonal matrix D_k and $\hat{m}$ pairs of vectors

$$\{(\mathbf{y}_j, \mathbf{s}_j) : k - \hat{m} \leq j \leq k - 1\} \tag{2.65}$$

if $k \geq \hat{m} + 1$, or just k pairs otherwise. In this way, at the k^{th} iteration with $k \geq \hat{m} + 1$, the oldest pair is discarded and a new pair added. The matrix G_k can be represented using $(2\hat{m} + 1)$ n-vectors, where $\hat{m}$ is an integer supplied by the user, and this is all that need be stored in memory.

The form of the starting matrix D_k has been found to be very important to the performance of quasi-Newton methods in general, and the paper by Oren and Spedicato [67] gives some detail on how D_k can be chosen. The N1QN3 algorithm (without the preconditioning option) specifies D_k to be the diagonal matrix

$$D_k = \delta_{k-1}I, \tag{2.66}$$

where the number δ_k is the *Oren-Spedicato factor*

$$\delta_{k-1} = \frac{(\mathbf{y}^{k-1})^T\mathbf{s}^{k-1}}{\|\mathbf{y}^{k-1}\|^2}, \tag{2.67}$$

which is intended to give G_k a good scaling.

The step-length ρ^k in (2.62) is chosen to satisfy *Wolfe's conditions*

$$\mathcal{J}(\mathbf{u}^{k+1}) \;\leq\; \mathcal{J}(\mathbf{u}^k) + \alpha_1 \rho^k < \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^k), G_k \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^k) >, \tag{2.68}$$

$$< \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^{k+1}), G_k \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^k) > \geq \;\alpha_2 < \nabla_{\mathbf{u}} \mathcal{J}(\mathbf{u}^k), G_k \nabla_{\mathbf{u}} (\mathbf{u}^k) >, \tag{2.69}$$

where the constants α_1 and α_2 must be set in the ranges $0 < \alpha_1 < \frac{1}{2}$ and $\alpha_1 < \alpha_2 < 1$. In the algorithm, these are set at the values $\alpha_1 = 10^{-4}$, $\alpha_2 = 0.9$.

The algorithm provides the option of preconditioning by altering the way in which D_k is specified in (2.66). In the preconditioned version, D_k is calculated from D_{k-1} using a diagonal update formula, and the matrix is now diagonal with respect to a new inner product to be specified by the user. This change of inner product is equivalent to a change of orthonormal basis from the canonical basis for $\mathbb{R}^n$, and this change of basis forms the preconditioning. If the usual inner product is the Euclidean product, as assumed in the above, then a new inner product could be of the form

$$< \mathbf{a}, \mathbf{b} >_L = \mathbf{a}^T L^T L \mathbf{b}, \tag{2.70}$$

where L is nonsingular, and the Canonical basis is altered by this change from the basis $\{\mathbf{e}_j\}$, $j = 1, .., n$ to the basis $\{L^{-1}\mathbf{e}_j\}$. Rather than storing the matrix L, or the new basis, the user provides a subroutine which specifies how the inner product is to be calculated.

2.5 Background on probability theory

This section gives a brief overview of probability theory. The aim is to introduce the concept of a statistically "most likely estimate", which is a very important concept in data assimilation. Before this, we give necessary definitions and background on the Gaussian distribution. References for this theory include [3], [14], [55] and [44].

2.5.1 Definitions

Random variables and probability density functions

A *random variable* can be thought of as a numerical value associated with a random event. The *range* of a random variable X, denoted R_X, is the set of all possible values

of X. We consider here only *continuous random variables*, or random variables with an uncountable range.

An n-vector $\mathbf{X}$ of random variables X_j, $j = 1, .., n$ we refer to as a *random n-vector*, or simply as a *random vector* if its dimension is not to be specified. The range of a random n-vector we denote $R_{\mathbf{X}}$, where $R_{\mathbf{X}} = R_{X_1} \times R_{X_2} \times ... \times R_{X_n}$.

Associated with any random variable X is a *probability density function* (abbreviated to pdf), $p_X : R_X \rightarrow \mathbb{R}$. The pdf of a continuous random variable X describes how the unit of probability of X is distributed on the real line. The probability $P(a \leq X \leq b)$ that X takes a value between a and $b \in \mathbb{R}$ is given by

$$P(a \leq X \leq b) = \int_a^b p_X(x)dx. \tag{2.71}$$

The other fundamental properties of a pdf are

$$p_X(x) \geq 0 \quad \text{for all} \quad x \in R_X, \tag{2.72}$$

$$\int_{R_X} p_X(x)dx = 1. \tag{2.73}$$

We write the pdf of a random n-vector as $\mathbf{p_X} : R_{\mathbf{X}} \rightarrow \mathbb{R}^n$, where $\mathbf{p_X}$ is the vector of the pdfs of the random variables X_j, $j = 1, .., n$.

The *joint pdf* of two random variables X and Y is given by $p_{XY} : R_X \times R_Y \rightarrow \mathbb{R}$, with

$$((a \leq X \leq b) \bigcap (c \leq Y \leq d)) = \int_a^b \int_c^d p_{XY}(x, y)\ dx\ dy, \tag{2.74}$$

$$p_{XY}(x, y) \geq 0 \quad \text{for all} \quad x \in R_X, y \in R_Y \tag{2.75}$$

$$\int_{R_X} \int_{R_Y} p_{XY}\ dx\ dy = 1 \tag{2.76}$$

The random variables X and Y are *independent* if

$$p_{XY}(x, y) = p_X(x)p_Y(y). \tag{2.77}$$

The *conditional* pdf of X, given that Y has taken a value y^0 (so y^0 is a *realisation* of the random variable Y) is defined to be

$$p_{X|Y=y^0}(x) = \frac{p_{XY}(x, y)}{p_Y(y^0)}. \tag{2.78}$$

This relation is from *Bayes theorem*, and it can also be written in the form

$$p_{X|Y=y^0}(x) = \frac{p_{Y|X=x}(y^0)p_X(x)}{p_Y(y^0)}. \tag{2.79}$$

We note that if X and Y are independent, then

$$p_{X|Y=y^0}(x) = p_X(x), \tag{2.80}$$

ie, knowledge that Y has taken a particular value has no impact on the probability of X.

Mean, mode, variance, covariance and correlation

The *mean value* or *expected value* $\mathcal{E}\{X\}$ of a random variable X is defined to be

$$\mathcal{E}\{X\} = \int_{-\infty}^{\infty} x p_X(x) dx. \tag{2.81}$$

A random variable is *unbiased* if $\mathcal{E}\{X\} = 0$. The mean of a random n-vector $\mathbf{X}$ is the vector of mean values of the components of $\mathbf{X}$,

$$\mathcal{E}\{\mathbf{X}\} = \begin{pmatrix} \mathcal{E}\{X_1\} \\ \vdots \\ \mathcal{E}\{X_n\} \end{pmatrix}. \tag{2.82}$$

The *expectation operator* $\mathcal{E}\{\ \}$ is a linear operator, and so has the following property for random vectors $\mathbf{X}$ and $\mathbf{Y}$,

$$\mathcal{E}\{A\mathbf{X} + B\mathbf{Y} + \mathbf{c}\} = A\mathcal{E}\{\mathbf{X}\} + B\mathcal{E}\{\mathbf{Y}\} + \mathbf{c}, \tag{2.83}$$

where A and B are constant matrices and $\mathbf{c}$ is a constant vector. The *mode* of a random variable is defined to be the value for which its pdf achieves a maximum.

The *variance* of a random variable X is defined to be

$$\mathrm{Var}\{X\} = \mathcal{E}\{(X - \mathcal{E}\{X\})^2\}, \tag{2.84}$$

which can be interpreted as the expected square distance from the mean. For constants a and $b \in \mathbb{R}$ we have

$$\mathrm{Var}\{aX + b\} = a^2 \mathrm{Var}\{X\}. \tag{2.85}$$

The *standard deviation* of X is defined as

$$\sigma(X) = (\mathrm{Var}\{X\})^{\frac{1}{2}}. \tag{2.86}$$

The *covariance* of two random variables X and Y is defined as

$$\mathrm{Cov}\{X, Y\} = \mathcal{E}\{(X - \mathcal{E}\{X\})(Y - \mathcal{E}\{Y\})\}, \tag{2.87}$$

and the *correlation* between X and Y is

$$\mathrm{Cor}\{X, Y\} = \mathcal{E}\{XY\}. \tag{2.88}$$

If $\mathrm{Cov}\{X, Y\} = 0$, then X and Y are *uncorrelated*. The *correlation coefficient* of X and Y is

$$\rho(X, Y) = \frac{\mathrm{Cov}\{X, Y\}}{\sigma(X)\sigma(Y)}, \tag{2.89}$$

where $-1 \leq \rho(X, Y) \leq 1$.

By linearity of the expectation operator, we have for matrices A and B of suitable dimensions

$$\mathrm{Cov}\{A\mathbf{X}, B\mathbf{Y}\} = A\mathrm{Cov}\{\mathbf{X}, \mathbf{Y}\}B^T. \tag{2.90}$$

We also note that if $\mathbf{X}$ and $\mathbf{Y}$ are unbiased, then

$$\mathrm{Cov}\{\mathbf{X}, \mathbf{Y}\} = \mathcal{E}\{\mathbf{X}\mathbf{Y}^T\}. \tag{2.91}$$

By the covariance matrix of a random vector $\mathbf{X}$, we mean the covariance matrix $\mathrm{Cov}\{\mathbf{X}, \mathbf{X}\}$.

2.5.2 The Gaussian distribution

We now suppose that a random variable X represents random error. The *Gaussian* distribution, also called the *Normal* distribution, has the following characteristics that make it suitable for representing errors:

1. Continuity

2. An unbounded range

3. Symmetry about the mean (so positive and negative errors are equally likely)

4. A "bell-shaped" distribution, which gives small probability to large errors and largest probability to the smallest errors,

5. Tractability, ie a pdf that is easy to work with.

The Gaussian pdf for a random variable X with mean μ and variance σ^2 is given by

$$p_X(x) = \frac{1}{\sqrt{(2\pi)\sigma}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right). \tag{2.92}$$

For a random n-vector with mean $\boldsymbol{\mu} \in \mathbb{R}^n$ and nonsingular covariance matrix R, the Gaussian pdf is

$$\mathbf{p_X}(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{n}{2}}(\det(R))^{\frac{1}{2}}} \exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T R^{-1}(\mathbf{x}-\boldsymbol{\mu})). \tag{2.93}$$

2.5.3 "Most likely" estimates

The following development broadly follows that of the paper by Lorenc [55].

We suppose that $\mathbf{x}^f$ is a "prior"estimate of a random n-vector $\mathbf{X}$. If we know that

$$\mathbf{X} = \mathbf{x}^f + \mathbf{e}^f, \tag{2.94}$$

where $\mathbf{e}^f$ is a random n-vector of the error $\mathbf{X} - \mathbf{x}^f$, and we know that $\mathbf{e}^f$ is Gaussian and unbiased with nonsingular covariance matrix P^f, then the pdf of $\mathbf{X}$ is given by

$$\mathbf{p_X}(\mathbf{x}) = k_1 \exp(-\frac{1}{2}(\mathbf{x}-\mathbf{x}^f)^T (P^f)^{-1}(\mathbf{x}-\mathbf{x}^f)), \tag{2.95}$$

where k_1 is a constant. We now suppose that we have a random p-vector $\mathbf{Y}$ that satisfies

$$\mathbf{Y} = C\mathbf{X} + \boldsymbol{\delta}, \tag{2.96}$$

where $C \in \mathbb{R}^{p \times n}$, and $\boldsymbol{\delta}$ is a random p-vector of the error $\mathbf{Y} - C\mathbf{X}$, which is Gaussian and unbiased, with nonsingular covariance matrix R. The conditional pdf for $\mathbf{Y}$ given that $\mathbf{X} = \mathbf{x}$ is

$$\mathbf{p_{Y|X=x}}(\mathbf{y}) = k_2 \exp(-\frac{1}{2}(C\mathbf{x}-\mathbf{y})^T R^{-1}(C\mathbf{x}-\mathbf{y})), \tag{2.97}$$

where k_2 is a constant. We now suppose that we have a particular realisation, $\mathbf{y}^0$ of $\mathbf{Y}$, and that we wish to find the "most likely" estimate $\mathbf{x}^a$ of $\mathbf{X}$ given that $\mathbf{Y} = \mathbf{y}^0$.

To do this we need to know $\mathbf{p}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^0}(\mathbf{x})$, which by (2.79) is given by

$$\mathbf{p}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^0}(\mathbf{x}) = \frac{\mathbf{p}_{\mathbf{Y}|\mathbf{X}=\mathbf{x}}(\mathbf{y}^0)\mathbf{p}_{\mathbf{X}}(\mathbf{x})}{\mathbf{p}_{\mathbf{y}}(\mathbf{y}^0)}, \tag{2.98}$$

hence

$$\mathbf{p}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^0}(\mathbf{x}) = \frac{k_1 k_2}{k_3} \exp(-\frac{1}{2}\{(C\mathbf{x}-\mathbf{y}^0)^T R^{-1}(C\mathbf{x}-\mathbf{y}^0)+(\mathbf{x}-\mathbf{x}^f)^T (P^f)^{-1}(\mathbf{x}-\mathbf{x}^f)\}), \tag{2.99}$$

where $k_3 = \mathbf{p}_{\mathbf{Y}}(\mathbf{y}^0)$ is a constant since $\mathbf{y}^0$ is given.

The "most likely estimate" of $\mathbf{x}^a$ could be defined either as the mode or the mean value of $\mathbf{X}$, which correspond to the *maximum likelihood* and *minimum variance* estimates respectively. Here it turns out that the maximum likelihood and minimum variance estimates coincide [55], and are given by $\mathbf{x}$ which maximises (2.99). Maximizing (2.99) is equivalent to minimizing the function

$$\mathcal{J}(\mathbf{x}) = \frac{1}{2}(C\mathbf{x}-\mathbf{y}^0)^T R^{-1}(C\mathbf{x}-\mathbf{y}^0) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^f)^T (P^f)^{-1}(\mathbf{x}-\mathbf{x}^f), \tag{2.100}$$

(since $\mathcal{J}(\mathbf{x}) = -\ln\frac{k_3}{k_1 k_2}\mathbf{p}_{\mathbf{X}|\mathbf{Y}=\mathbf{y}^0}(\mathbf{x})$).

In summary, if $\mathbf{X}$ and $\mathbf{Y}$ are random vectors satisfying (2.94) and (2.96), then the most likely estimate of $\mathbf{X}$ given that $\mathbf{Y} = \mathbf{y}^0$ and a prior estimate $\mathbf{X} = \mathbf{x}^f$, is given by $\mathbf{x}^a \in \mathbb{R}^n$ which minimizes $\mathcal{J}$ in (2.100).

Chapter 3

Sequential data assimilation

Sequential data assimilation schemes treat observations as they become available in time, and then discard them. If a 3D data assimilation method, which is designed to produce an analysis at a single time, is applied repeatedly, this can be seen as sequential data assimilation. 4D sequential data assimilation methods, however, are designed so that an analysis should gradually draw closer to the true model state, as more observations are processed. In control theory, dynamic observers are designed for this very purpose, and so observer theory is very relevant to sequential data assimilation. An example of an observer originally designed for engineering applications which is being investigated for use in data assimilation, is the Kalman filter. The Kalman filter is designed to produce a solution that is, under certain assumptions, statistically optimal. There are also other ways of designing observers which give the solution other desirable properties.

In this chapter, we firstly give a brief outline of some 3D data assimilation schemes. In Section 3.2, we give an introduction to the Kalman filter. We pay particular attention to the assumptions made on observational error and model error, and how the Kalman filter can be generalized to allow for serially correlated model error, since we use these ideas later in the thesis. Then, in Section 3.3, we give an introduction to dynamic observers of control theory, and give theory on design of a *robust* observer using eigenstructure assignment. In Section 3.4, we discuss how 3D data assimilation schemes can be extended to 4D schemes. We show how the successive correction method can be expressed as an observer if observations

are available frequently. Using observer theory, we are able to give conditions for the linear, time invariant case under which the successive correction analysis will converge *in time* to the true solution. In Section 3.5, we compare the Cressman successive correction scheme with a robust observer in data assimilation for a simple example. These experiments serve to illustrate how an observer which is designed for temporal convergence to the true solution can perform much better than successive correction scheme designed for an analysis at a single time.

3.1 Background on 3D data assimilation schemes

By "3D" data assimilation schemes we mean schemes that are designed to give an analysis at a single time, and do not attempt to take into account the time-tendency of the observations. This section gives a brief overview of a few 3D data assimilation schemes that have been used in the past and to date, and which we use or refer to in this thesis. We firstly outline successive correction methods, which are some of the earlier schemes to have been proposed and implemented. We then introduce the method of optimal interpolation, on which the schemes currently used in many meteorological centres are based. Finally, we describe the 3D variational assimilation (3DVAR) method which is being developed for operational use at several centres as an intermediate stage in the development of 4D variational assimilation (4DVAR) schemes.

The material in this section is intended to be only a brief outline of the methods discussed. A more in-depth overview of data assimilation methods and further references are given in the review paper by Ghil and Malenotte-Rissoli [32] and the books by Daley [24] and Bennett [4].

3.1.1 Successive correction schemes

Successive correction schemes were introduced to meteorology in the 1950s for operational objective analysis, by Bergthórson and Döös [9], and by Cressman [21]. The *Cressman scheme* [21] was designed for systems with few observations, widely scattered, which are to be fitted as closely as possible. This method was intend-

ed to improve on the earlier polynomial spline methods [68], [35] by being more suitable for use over larger areas with less dense data coverage, and by being computationally simpler [21]. Successive correction schemes have been widely used in data assimilation, [24].

We suppose we have a prior estimate $\mathbf{x}_k^b$ of the true model state $\mathbf{x}_k^t$, and observations given by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t), \tag{3.1}$$

where equation (3.1) is as defined in (2.6) assuming no observational error. The successive correction method is an iteration on $\mathbf{x}_k$ which brings it successively closer to the observations $\mathbf{y}_k$. This iteration has the following general form, [55],

$$\mathbf{x}_k^{(i+1)} = \mathbf{x}_k^{(i)} + QW^{(i+1)}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^{(i)})), \qquad i = 0, .., s-1, \tag{3.2}$$

with $\mathbf{x}_k^{(0)} = \mathbf{x}_k^b$, where $\mathbf{x}_k^{(i)}$ represents the i^{th} iterate of $\mathbf{x}_k$, $W^{(i)} \in \mathbb{R}^{n \times p_k}$ are weighting matrices, $Q \in \mathbb{R}^{n \times n}$ contains normalizing factors, and s is the total number of iterations. The analysed state is then given by $\mathbf{x}_k^a = \mathbf{x}_k^{(s)}$. The weighting matrix in effect smoothes the observational data into the model state by modifying the state at grid points within some *radius of influence* of each observation point. Although the weights in a successive correction method are generally empirically determined, some methods, including the original method by Bergthorsen and Döös, use the statistics of analysis error to determine the weights, and so are able to allow for observational error [24]. The paper by Lorenc [55] shows how the successive correction methods can be related to statistically optimal methods.

The Cressman scheme

The Cressman scheme [21] is one of the earliest schemes for objective analysis to have been used operationally, and has been widely used since [24]. We give more detail on this method, since we implement it for a simple example in Section 3.5. In this method, the iteration is repeated with successively smaller radii of influence $R^{(i)}$, which has the effect of altering the large scale features of the motion on the first iterations, and the smaller scale features on successive corrections [24]. The

$(l,m)^{th}$ element of the matrix $W^{(i)}$ is given by

$$W_{lm}^{(i)} = \frac{(R_m^{(i)})^2 - d_{lm}^2}{(R_m^{(i)})^2 + d_{lm}^2}, \qquad l = 1,..,n, \quad m = 1,..,p_k, \tag{3.3}$$

where $R_m^{(i)}$ is the radius of influence at the i^{th} iteration for observation m (ie, the m^{th} component of $\mathbf{y}_k$), and d_{lm} is the distance between observation m and grid point l. In the original paper introducing this method, 4 iterations were carried out with different radii of influence [21].

3.1.2 Optimal interpolation

The method of optimal interpolation (OI) has been widely used in operational data assimilation for NWP in the 1980s and 1990s [42]. Important references for the method include the papers by Gandin [30] and Lorenc [55]. The OI method was designed for a system in which observations are linearly related to the model state. We suppose that we wish to estimate $\mathbf{x}_k^t$, and that we have observations given by

$$\mathbf{y}_k = C_k \mathbf{x}_k^t + \boldsymbol{\delta}_k, \tag{3.4}$$

where $\mathbf{y}_k$ and $\boldsymbol{\delta}_k$ are defined as in (2.9). We suppose that the observational error $\boldsymbol{\delta}_k$ is an unbiased, Gaussian random vector, with nonsingular covariance matrix R_k. We also suppose that we have a prior estimate (or "background estimate") $\mathbf{x}_k^b$ of $\mathbf{x}_k^t$, and that the error $(\mathbf{x}_k^t - \mathbf{x}_k^b)$ is an unbiased, Gaussian random vector with nonsingular covariance matrix P_k. The OI method is based on finding the most likely state $\mathbf{x}_k^a$ at time t_k from the prior estimate $\mathbf{x}_k^b$ and the vector of observations $\mathbf{y}_k$. From Chapter 2, Section 2.5, we have that the most likely estimate minimizes

$$\mathcal{J}(\mathbf{x}_k) = \frac{1}{2}(\mathbf{x}_k - \mathbf{x}_k{}^b)^T P_k^{-1}(\mathbf{x}_k - \mathbf{x}_k{}^b) + \frac{1}{2}(C_k\mathbf{x}_k - \mathbf{y}_k)^T R_k^{-1}(C_k\mathbf{x}_k - \mathbf{y}_k). \tag{3.5}$$

The OI analysis $\mathbf{x}_k^a$ satisfies

$$\mathbf{x}_k{}^a = \mathbf{x}_k^b + W_k(\mathbf{y}_k - C_k\mathbf{x}_k^b), \tag{3.6}$$

where the OI weighting matrix W_k is specified by

$$W_k = P_k C_k^T (C_k P_k C_k^T + R_k)^{-1}, \tag{3.7}$$

as discussed in [56]. The OI method is in fact not truly optimal since it does not update the error covariance matrix P_k of the background estimate in a way that takes into account the earlier observations which have already been assimilated. The Kalman filter does this, but the extra cost involved is large. The OI method is sometimes more realistically referred to as *statistical interpolation*, [54]. Although designed for a linear system, the method can be extended for use in a system in which the observations are nonlinearly related to the model state,

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \tag{3.8}$$

with $\mathbf{y}_k$, $\mathbf{x}_k^t$ and $\boldsymbol{\delta}_k$ as defined in (3.4), and where $\mathbf{h}_k : \mathbb{R}^n \rightarrow \mathbb{R}^{p_k}$ is a nonlinear operator. This can be done by linearizing (3.8) about $\mathbf{x}_k^b$. We describe this approach in a little more detail in the context of the nonlinear extension to the Kalman filter in Section 3.2.

3.1.3 3D variational assimilation, and the PSAS method

The three dimensional variational assimilation (3DVAR) method takes a different approach to minimizing the function (3.5). Rather than solving equations (3.6) and (3.7), the approach is to iterate to the minimizing solution $\mathbf{x}_k^a$. The gradient of (3.5) with respect to $\mathbf{x}_k$ is

$$\nabla_{\mathbf{x}_k}\mathcal{J} = P_k^{-1}(\mathbf{x}_k - \mathbf{x}_k^b) + C_k^T R_k^{-1}(C_k\mathbf{x}_k - \mathbf{y}_k), \tag{3.9}$$

and this may be used in a gradient method to iterate to the optimal solution. We describe a few such methods in Chapter 2, Section 2.4.

The 3DVAR method is currently being developed for implementation for operational data assimilation at several meteorological centres, with plans for extension to the 4DVAR method [73]. The "PSAS" method, or *physical-space statistical analysis system* [42] represents another way of solving equations (3.6),(3.7). The approach taken in this case is to solve for $\mathbf{w}_k \in \mathbb{R}^{p_k}$ the equation

$$(C_k P_k C_k^T + R_k)\mathbf{w}_k = (\mathbf{y}_k - C_k\mathbf{x}_k^b) \tag{3.10}$$

using some suitable iterative method. The solution $\mathbf{x}_k^a$ is then given by

$$\mathbf{x}_k^a = \mathbf{x}_k^b + P_k C_k^T \mathbf{w}_k. \tag{3.11}$$

In general, the dimension p_k of the observation vector is much smaller than the dimension n of the model state $\mathbf{x}_k$; for meteorological applications it might be two orders of magnitude less, and even less for oceanographic applications. The advantage of the "physical-space" approach lies in this potential gain in efficiency.

The 3DVAR and PSAS methods may be extended to cases where the observations are nonlinearly related to the model state, as in equation (3.8), as we described for the OI method. Alternatively, the cost function $\mathcal{J}$ may be explicitly defined for the nonlinear system, as we describe in the context of 4D variational assimilation in Chapter 4.

3.2 The Kalman filter

The Kalman filter for a discrete linear model with observations linearly related to the model state was developed by Kalman in 1960 [45]. The continuous time version was developed by Kalman and Bucy in 1961 [46]. Here we concentrate on the discrete version. Comprehensive background on the discrete Kalman filter is given in the texts [14] and [44]. An introduction to the Kalman filter for data assimilation applications can be found, for example, in [60].

The Kalman filter has been considered for application in meteorology and oceanography, but is generally considered too expensive for operational implementation because of the large dimension of the problem [32]. However, several simplifications of the method have been suggested for data assimilation, [84]. Further, since the Kalman filter provides, for a linear system and under certain statistical assumptions, a statistically optimal solution for 4D data assimilation, it is useful to exploit links between the Kalman filter and other data assimilation methods. We describe the assumptions made in Kalman filtering on observational errors and model errors in some detail, since we refer to these same assumptions later in the thesis. We then describe the Kalman filter, and give some detail on how it can be modified to deal with serially correlated model error, since later in the thesis we discuss further how to deal with serially correlated model error in variational data assimilation. Finally, we discuss briefly how the Kalman filter can be used for nonlinear systems.

3.2.1 The standard Kalman filter assumptions

We suppose that the true model state is defined by the linear *stochastic dynamic* system

$$\mathbf{x}_{k+1}^t = A_k \mathbf{x}_k^t + B_k \mathbf{u}_k + \boldsymbol{\varepsilon}_k, \tag{3.12}$$

where $\mathbf{x}_k^t$ is a random n-vector representing the true model state at time t_k, $\mathbf{u}_k \in \mathbb{R}^m$ is a vector of specified model inputs, $A_k \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times m}$, and $\boldsymbol{\varepsilon}_k$ is a Gaussian random n-vector representing *model error*, with nonsingular covariance matrix $Q_k \in \mathbb{R}^{n\times n}$. We suppose that the output of (3.12) at time t_k is a random p_k-vector $\mathbf{y}_k$ related to the state $\mathbf{x}_k^t$ by

$$\mathbf{y}_k = C_k \mathbf{x}_k^t + \boldsymbol{\delta}_k, \tag{3.13}$$

where $C_k \in \mathbb{R}^{p_k\times n}$ and $\boldsymbol{\delta}_k$ is a Gaussian random p_k-vector representing *observational error*, with non-singular covariance matrix $R_k \in \mathbb{R}^{p_k\times p_k}$. We suppose that we have a prior estimate, called a *forecast* in this context, $\mathbf{x}_k^f$ of $\mathbf{x}_k^t$, and that the *forecast error* at time t_k, defined to be

$$\mathbf{e}_k^f = \mathbf{x}_k^f - \mathbf{x}_k^t, \tag{3.14}$$

is a Gaussian random n-vector with zero mean and with nonsingular covariance matrix P_k^f.

In the standard Kalman filter, the following assumptions are made about the model error $\boldsymbol{\varepsilon}_k$,

ME1 : it is unbiased, $\mathcal{E}\{\boldsymbol{\varepsilon}_k\} = 0$,

ME2 : it is serially uncorrelated (white), ie $\mathrm{Cov}\{\boldsymbol{\varepsilon}_k, \boldsymbol{\varepsilon}_j\} = 0, \quad j \neq k$.

We make similar assumptions about the observational error $\boldsymbol{\delta}_k$,

OE1 : it is unbiased, $\mathcal{E}\{\boldsymbol{\delta}_k\} = 0$,

OE2 : it is serially uncorrelated, ie $\mathrm{Cov}\{\boldsymbol{\delta}_k, \boldsymbol{\delta}_j\} = 0 \quad j \neq k$.

It is further assumed that model error and observational error are uncorrelated with each other and uncorrelated with $\mathbf{x}_0^f$,

MOE :

$$\mathrm{Cov}\{\boldsymbol{\varepsilon}_k, \boldsymbol{\delta}_j\} = 0, \qquad \mathrm{Cov}\{\boldsymbol{\varepsilon}_k, \mathbf{x}_0^f\} = 0, \qquad \mathrm{Cov}\{\boldsymbol{\delta}_k, \mathbf{x}_0^f\} = 0, \quad \forall j, k. \tag{3.15}$$

We note that, from (3.12) and ME2, we have

$$\mathrm{Cov}\{\mathbf{x}_j^t, \boldsymbol{\varepsilon}_k\} = 0, \qquad \forall k \geq j, \tag{3.16}$$

and similarly, from (3.13) and OE2 we have

$$\mathrm{Cov}\{\mathbf{x}_j^t, \boldsymbol{\delta}_k\} = 0, \qquad \forall k > j. \tag{3.17}$$

As mentioned above, we assume that the error covariance matrices Q_k, R_k and P_k^f are nonsingular. All covariance matrices for a random vector with itself are symmetric positive semi-definite, and are positive definite if they do not contain null variances or perfect correlations [81]. We assume that this is so. Assumption ME1 that model error is unbiased is in fact not restrictive; if we have

$$\mathcal{E}\{\boldsymbol{\varepsilon}_k\} = \bar{\boldsymbol{\varepsilon}_k} \neq 0, \tag{3.18}$$

we can define

$$\boldsymbol{\varepsilon}_k = \bar{\boldsymbol{\varepsilon}_k} + \boldsymbol{\varepsilon}_k{}' \tag{3.19}$$

where $\mathcal{E}\{\boldsymbol{\varepsilon}_k{}'\} = 0$ and $\bar{\boldsymbol{\varepsilon}_k} \in \mathbb{R}^n$ is now part of the deterministic model forcing, and continue as before. Similarly, assumption OE1 on observational error is not restrictive. Assumptions ME2 and OE2 can also be relaxed, but doing so necessitates extra computational cost and extra statistical information.

3.2.2 The Kalman filter

The Kalman filter finds the most likely estimate $\mathbf{x}_k^a$ of the state $\mathbf{x}_k^t$ from a prior estimate $\mathbf{x}_k^f$ and a vector $\mathbf{y}_k^0$ of observations, which is a realization (or particular outcome) of the random vector $\mathbf{y}_k$. The vector $\mathbf{x}_k^a$ is sometimes called the *analysis*. From Chapter 2, Section 2.5, we know that $\mathbf{x}_k^a$ minimizes the function

$$\mathcal{J}(\mathbf{x}_k) = \frac{1}{2}(\mathbf{x}_k - \mathbf{x}_k^f)(P_k^f)^{-1}(\mathbf{x}_k - \mathbf{x}_k^f) + \frac{1}{2}(C_k\mathbf{x}_k - \mathbf{y}_k^0)^T R_k^{-1}(C_k\mathbf{x}_k - \mathbf{y}_k^0). \tag{3.20}$$

A necessary condition for a minimum is that $\nabla_{\mathbf{x}_k} \mathcal{J}(\mathbf{x}_k) = 0$, ie

$$(P_k^f)^{-1}(\mathbf{x}_k - \mathbf{x}_k^f) + C_k^T R_k^{-1}(C_k \mathbf{x}_k - \mathbf{y}_k^0) = 0. \tag{3.21}$$

It can be shown [60] that the best estimate $\mathbf{x}_k^a$ is a unique, global minimum of (3.21), and satisfies

$$\mathbf{x}_k^a = \mathbf{x}_k^f + K_k(\mathbf{y}_k^0 - C_k \mathbf{x}_k^f) \tag{3.22}$$

where $K_k \in \mathbb{R}^{n \times p_k}$ is the *Kalman gain matrix* given by

$$K_k = P_k^f C_k^T (C_k P_k^f C_k^T + R_k)^{-1}. \tag{3.23}$$

We define the *analysis error* at time t_k to be

$$\mathbf{e}_k^a = \mathbf{x}_k^a - \mathbf{x}_k^t. \tag{3.24}$$

Subtracting $\mathbf{x}_k^t$ from (3.22) and using (3.13) gives the following equation for $\mathbf{e}_k^a$,

$$\begin{aligned} \mathbf{e}_k^a &= \mathbf{e}_k^f + K_k(C_k \mathbf{x}_k^t + \boldsymbol{\delta}_k - C_k \mathbf{x}_k^f) && (3.25) \\ &= (I - K_k C_k)\mathbf{e}_k^f + K_k \boldsymbol{\delta}_k. && (3.26) \end{aligned}$$

Since $\mathbf{e}_k^f$ and $\boldsymbol{\delta}_k$ are unbiased, $\mathbf{e}_k^a$ is also unbiased. Further, it can be verified that $\mathbf{e}_k^f$ and $\boldsymbol{\delta}_k$ are uncorrelated because of equation (3.17), and hence $\boldsymbol{e}_k^a$ has covariance matrix

$$P_k^a = (I - K_k C_k) P_k^f (I - K_k C_k)^T + K_k R_k K_k^T, \tag{3.27}$$

which can also be written [60]

$$P_k^a = (I - K_k C_k) P_k^f. \tag{3.28}$$

So far, we have specified the optimal analysis $\mathbf{x}_k^a$ at time t_k and its error covariance matrix P_k^a. We now move on to the next step, and calculate the prior estimate or forecast of $\mathbf{x}_{k+1}^t$ as follows

$$\mathbf{x}_{k+1}^f = A_k \mathbf{x}_k^a + B_k \mathbf{u}_k. \tag{3.29}$$

The forecast error covariance matrix P_{k+1}^f must now be calculated. Subtracting (3.12) from (3.29) gives the following expression

$$\mathbf{e}_{k+1}^f = A_k \mathbf{e}_k^a - \boldsymbol{\varepsilon}_k. \tag{3.30}$$

It can be verified that $\mathbf{e}_k^a$ and $\boldsymbol{\varepsilon}_k$ are uncorrelated because of the relation (3.16). Further, $\mathbf{e}_{k+1}^f$ is unbiased since $\mathbf{e}_k^a$ and $\boldsymbol{\varepsilon}_k$ are, and we have

$$P_{k+1}^f = A_k P_k^a A_k^T + Q_k. \tag{3.31}$$

We can now apply the Kalman filter equations (3.22) and (3.23) to find $\mathbf{x}_{k+1}^a$, which is the best estimate of $\mathbf{x}_{k+1}^t$ given the observation vector $\mathbf{y}_{k+1}^0$ and prior estimate $\mathbf{x}_{k+1}^f$, and (3.28) to find its error covariance matrix.

In fact, since $\mathbf{x}_{k+1}^f$ is the best estimate of $\mathbf{x}_{k+1}^t$ from the observation vector $\mathbf{y}_k^0$ and prior estimate $\mathbf{x}_k^f$, then $\mathbf{x}_{k+1}^a$ is the best estimate of $\mathbf{x}_{k+1}^t$ from the observations $\mathbf{y}_k^0$ and $\mathbf{y}_{k+1}^0$ and prior estimate $\mathbf{x}_k^f$. If observations $\mathbf{y}_0^0$, $\mathbf{y}_1^0$,...$\mathbf{y}_{k+1}^0$ have been treated in this way, then $\mathbf{x}_{k+1}^a$ is the best estimate of $\mathbf{x}_{k+1}^t$ given observations $\mathbf{y}_0^0$,...,$\mathbf{y}_{k+1}^0$ and prior estimate $\mathbf{x}_0^f$ [44].

3.2.3 Serially correlated model error

If we wish to allow for serial correlation in model error, and abandon assumption ME2, then the Kalman filtering equations can be modified in the way described below. In Chapter 6 we give a fuller discussion on why this modification might be needed. The following theory broadly follows a paper by Daley written in the context of data assimilation [23], and the book by Jazwinski [44], in which the case of serially correlated observation error rather than model error is treated.

To account for serially correlated model error in the system (3.12),(3.13), we must assume that the serial correlation is known. We therefore suppose that we have the following linear stochastic dynamic model for the evolution of model error,

$$\boldsymbol{\varepsilon}_{k+1} = G_k \boldsymbol{\varepsilon}_k + \mathbf{q}_k \tag{3.32}$$

where $G_k \in \mathbb{R}^{n\times n}$ represents the dynamic evolution of model error from time t_k to t_{k+1}, and $\mathbf{q}_k$ is an unbiased, Gaussian random n-vector, which is assumed to be serially uncorrelated, and uncorrelated with the observational error. We assume that $\mathbf{q}_k$ has a nonsigular covariance matrix S_k. We note that from (3.32) and the fact that $\mathbf{q}_k$ is serially uncorrelated, we have

$$\mathrm{Cov}\{\boldsymbol{\varepsilon}_j, \mathbf{q}_k\} = 0, \qquad \forall k \geq j. \tag{3.33}$$

The other assumptions on model error and observational error are as before, including the assumption that model error and observational error are uncorrelated.

The analysis given for the standard Kalman filter in equations (3.22) and (3.23) still holds for producing the best estimate or analysis $\mathbf{x}_k^a$ based on $\mathbf{x}_k^f$, $\mathbf{y}_k^0$ and their error covariance matrices. The expressions (3.27) and (3.28) for the analysis error covariance are also unchanged, since the forecast error and observational error are still uncorrelated. The expression for the new forecast error $\mathbf{e}_{k+1}^f$ given in (3.30) is also unchanged, but because model error is now serially correlated, equation (3.16) no longer holds, and model error and analysis error are now correlated. Hence, the expression (3.31) for the new forecast error covariance matrix must be modified as follows.

$$\begin{aligned} P_{k+1}^f &= \mathrm{Cov}\{(A_k\mathbf{e}_k^a - \boldsymbol{\varepsilon}_k), (A_k\mathbf{e}_k^a - \boldsymbol{\varepsilon}_k)\} \\ &= A_k P_k^a A_k^T - A_k\mathrm{Cov}\{\mathbf{e}_k^a, \boldsymbol{\varepsilon}_k\} - \mathrm{Cov}\{\mathbf{e}_k^a, \boldsymbol{\varepsilon}_k\}A_k^T + Q_k, \end{aligned} \tag{3.34}$$

or, defining the covariance matrix of analysis and model errors as

$$P_k^{aq} = \mathrm{Cov}\{\mathbf{e}_k^a, \boldsymbol{\varepsilon}_k\}, \tag{3.35}$$

we have

$$P_{k+1}^f = A_k P_k^a A_k^T - A_k P_k^{aq} - P_k^{aq} A_k^T + Q_k \tag{3.36}$$

It now remains to specify P_{k+1}^{aq} from P_k^{aq}. This is given by

$$P_{k+1}^{aq} = (I - K_k C_k)(A_k P_k^{aq} - Q_k)G_k^T. \tag{3.37}$$

Finally, Q_{k+1} is calculated from Q_k as follows

$$Q_{k+1} = G_k Q_k G_k^T + S_k. \tag{3.38}$$

So, for serially correlated model error, we have the standard Kalman filter equations, except that the evolution of the forecast error covariance (3.31) is modified to (3.36), and in addition we must propagate the covariance matrix of analysis and model errors as expressed in equation (3.37), and the model error covariance matrix (3.38).

We note that serially correlated observational errors can be dealt with in a similar way, if we assume model error is uncorrelated in time [23]. In this case the equation

for the analysis error covariance propagation would be modified, and we would need to work out the propagation of the covariance matrix of forecast and observational errors. It is also possible to allow for serial correlations in both model error and observational error [44], but at greater complication still.

3.2.4 The extended Kalman filter

We now consider the extension of Kalman filtering theory to the nonlinear, stochastic dynamic system

$$\mathbf{x}_k^t = \mathbf{f}_k(\mathbf{x}_k^t, \mathbf{u}_k) + \boldsymbol{\varepsilon}_k, \tag{3.39}$$

with observations nonlinearly related to the state as follows

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \tag{3.40}$$

where the true state $\mathbf{x}_k^t$, the output $\mathbf{y}_k$, the specified input $\mathbf{u}_k$ and random errors $\boldsymbol{\varepsilon}_k$ and $\boldsymbol{\delta}_k$ are defined as in the system (3.12),(3.13), and $\mathbf{f}_k : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\mathbf{h}_k : \mathbb{R}^n \to \mathbb{R}^{p_k}$ are nonlinear operators.

There are various ways of developing an extended Kalman filter (EKF) for the system (3.39),(3.40), [44]. Here we give a fairly brief treatment of the subject, using one approach which is popular in the context of data assimilation for meteorology and oceanography, [32]. This approach involves linearizing the system about the forecast state $\mathbf{x}_k^f$, to obtain a linear system of the form

$$\mathbf{x}'_{k+1} = A_k(\mathbf{x}_k^f, \mathbf{u}_k)\mathbf{x}'_k + B_k(\mathbf{x}_k^f, \mathbf{u}_k)\mathbf{u}'_k + \boldsymbol{\varepsilon}'_k, \tag{3.41}$$

$$\mathbf{y}'_k = C_k(\mathbf{x}_k^f)\mathbf{x}'_k + \boldsymbol{\delta}'_k, \tag{3.42}$$

in which $A_k \in \mathbb{R}^{n\times n}$ and $C_k \in \mathbb{R}^{p_k\times n}$ are the Jacobians with respect to $\mathbf{x}_k$ of $\mathbf{f}_k$ and $\mathbf{h}_k$ respectively, and $\mathbf{B}_k \in \mathbb{R}^{n\times m}$ is the Jacobian of $\mathbf{f}_k$ with respect to $\mathbf{u}_k$, and the errors $\boldsymbol{\varepsilon}'_k$ and $\boldsymbol{\delta}'_k$ are assumed to satisfy the standard Kalman filter assumptions ME1,ME2, OE1,OE2, and MOE. In the EKF, the nonlinear evolution

$$\mathbf{x}_{k+1}^f = \mathbf{f}_k(\mathbf{x}_k^a, \mathbf{u}_k) \tag{3.43}$$

replaces (3.29), but the analysis (3.22) with (3.23) and the evolution of the covariance matrices given in (3.28) and (3.31) are carried out using the linearized

system (3.41),(3.42). In this case, the equation for the forecast error covariance propagation is correct to first order in $\mathbf{e}_k^f$.

3.3 Observer theory

In control theory, a *dynamic observer* uses model outputs (observations) to drive a model state closer to the true state as characterised by the observations. For this reason the sequential data assimilation methods can be expressed quite naturally in terms of observers, because the approach in sequential assimilation is to gradually drive the model state closer to the optimal solution over the assimilation interval.

The Kalman filter is an example of a stochastic observer, although in this case the aim is not to drive the model to the observations exactly, but to the "most likely" model state given appropriate error covariance information. Other forms of observer can be formulated for data assimilation and a class of observers known as "simplified Kalman filters" (SKF) are being developed for this application [84], [32]. If a 3D method of data assimilation, such as a successive correction method, is applied repeatedly, it could also be expressed in terms of observers, as we discuss in Section 3.4.

We begin this section by introducing a dynamic observer for a nonlinear system, and looking for criteria for its convergence to the true solution. In the case where we restrict our attention to the linear, time invariant system, it is easy to express conditions for convergence in terms of the eigenvalues of the observer. We discuss how, under certain conditions, we can design the eigenstructure of the observer system so that it behaves in a desirable way. Then we describe a method of observer design that results in a robust observer system, which serves as an example of observer design.

3.3.1 Dynamic observers

We consider the nonlinear model

$$\mathbf{x}_{k+1}^t = \mathbf{f}_k(\mathbf{x}_k^t, \mathbf{u}_k) \tag{3.44}$$

with observations

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) \tag{3.45}$$

defined as in (2.1),(2.6), assuming no model error or observation error. A dynamic observer for (3.44),(3.45) may be written in the form

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k, \mathbf{u}_k) + G_k(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)), \tag{3.46}$$

where $\mathbf{x}_k$ is an estimate of the true model state $\mathbf{x}_k^t$, and the *feedback matrix* $G_k \in \mathbb{R}^{n \times p_k}$ must be chosen so that $\mathbf{x}_k \to \mathbf{x}_k^t$ as $t_k \to \infty$.

Expanding (3.46) in a Taylors series about $\mathbf{x}_k^t$, and defining $\mathbf{e}_k = \mathbf{x}_k - \mathbf{x}_k^t$ we have

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k^t, \mathbf{u}_k) + F_k(\mathbf{x}_k^t, \mathbf{u}_k)\mathbf{e}_k + G_k(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^t) - H_k(\mathbf{x}_k^t)\mathbf{e}_k) + \mathbf{o}(\mathbf{e}_k), \tag{3.47}$$

where F_k and H_k are the Jacobians of $\mathbf{f}_k$ and $\mathbf{h}_k$ with respect to $\mathbf{x}_k$, and $\mathbf{o}(\mathbf{e}_k)$ represents the higher order terms. Now using (3.45) and subtracting (3.44) from (3.47) we have the following equation for the error $\mathbf{e}_k$ between the observer state and true state

$$\mathbf{e}_{k+1} = (F_k(\mathbf{x}_k, \mathbf{u}_k) - G_k H_{\mathbf{x}_k}(\mathbf{x}_k))\mathbf{e}_k + \mathbf{o}(\mathbf{e}_k). \tag{3.48}$$

The nonlinear observer (3.47) will converge to the true model state provided $\mathbf{e}_k \to 0$ as $t_k \to \infty$, with the evolution of $\mathbf{e}_k$ given by (3.48). An example of a nonlinear observer, which uses a gradient method with an interesting link to the 4D variational assimilation method, is given in [62].

The linear, time invariant case

For the linear, time invariant system

$$\mathbf{x}_{k+1}^t = A\mathbf{x}_k^t + B\mathbf{u}_k \tag{3.49}$$

$$\mathbf{y}_k = C\mathbf{x}_k^t, \tag{3.50}$$

as defined in (2.26),(2.27), a dynamic observer of the form (3.46) is

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{u}_k + G(\mathbf{y}_k - C\mathbf{x}_k), \tag{3.51}$$

and the error equation corresponding to (3.48) is

$$\mathbf{e}_{k+1} = (A - GC)\mathbf{e}_k, \tag{3.52}$$

and therefore

$$\mathbf{e}_k = (A - GC)^k \mathbf{e}_0. \tag{3.53}$$

Hence, to satisfy the condition $\mathbf{e}_k \to 0$ as $t_k \to \infty$, the eigenvalues of (A-GC), denoted by $\lambda_i(A - GC)$, must satisfy the condition

$$|\lambda_i(A - GC)| < 1 \qquad \forall i = 1, .., n. \tag{3.54}$$

In certain cases it is possible to choose the feedback matrix G so that the matrix $(A - GC)$ has any specified eigenvalues, and in particular, it is possible to ensure that the condition (3.54) holds. Theorem 3.1 gives sufficient conditions for this to hold [90].

Theorem 3.1 *If the system (3.49),(3.50) is completely observable, then it is possible to choose the matrix G in (3.51) so that the eigenvalues of $(A - GC)$ take prescribed values.*

3.3.2 Eigenstructure assignment

The *inverse eigenvalue problem* of assigning eigenvalues to the system (3.51) allows some freedom in choosing the corresponding eigenvectors in the case $p > 1$, and since we have some freedom in choosing the eigenvectors also, our problem now is one of *eigenstructure assignment* [28]. We now describe how we can choose the eigenstructure of the dynamic observer (3.51).

We suppose that the conditions of Theorem 3.1 hold, and that the set of eigenvalues we wish to assign is

$$\Lambda = \{\lambda_1, \lambda_2, ..., \lambda_n\}; \tag{3.55}$$

where

$$\lambda_i \in \mathbb{C}, \quad |\lambda_i| < 1, \quad \text{and} \quad \lambda \in \Lambda \Rightarrow \bar{\lambda} \in \Lambda \qquad \text{for} \quad i = 1, ..., n. \tag{3.56}$$

We let $D = diag\{\lambda_i\}$ and let X be the modal matrix of right eigenvectors of $(A-GC)$ and Y be the modal matrix of $(A^T - C^T G^T)$. Then our problem is to choose G and X to satisfy

$$(A - GC)X = XD, \tag{3.57}$$

or, equivalently, to choose Y and G^T to satisfy

$$(A^T - C^T G^T)Y = YD. \tag{3.58}$$

For our purposes, we work with equation (3.58).

If we calculate the QR decomposition of C^T, we find that

$$C^T = [\tilde{Q}_c, Q_c] \begin{bmatrix} R_o \\ 0 \end{bmatrix}, \tag{3.59}$$

where $\tilde{Q}_c \in \mathbb{R}^{n\times p}$, $Q_c \in \mathbb{R}^{n\times(n-p)}$; $[\tilde{Q}_c, Q_c]$ is orthogonal and $R_o \in \mathbb{R}^{p\times p}$ is upper triangular, nonsingular since C is assumed to have rank p. Substituting this into (3.58) and rearranging gives

$$\begin{pmatrix} \tilde{Q}_c^T A^T Y - \tilde{Q}_c^T YD \\ Q_c^T A^T Y - Q_c^T YD \end{pmatrix} = \begin{pmatrix} R_o G^T Y \\ 0 \end{pmatrix}, \tag{3.60}$$

from which we have

$$G^T = R_o^{-1} \tilde{Q}_c^T (A^T Y - YD) Y^{-1}, \tag{3.61}$$

$$0 = Q_c^T (A^T Y - YD). \tag{3.62}$$

Equation (3.61) is the equation for G^T for a given Y and equation (3.62) gives us a condition for choosing Y.

From (3.62) we have that for $i = 1, ..., n$

$$Q_c^T (A^T - \lambda_i I)\boldsymbol{\eta}_i = 0, \tag{3.63}$$

where $\boldsymbol{\eta}_i$ is the i^{th} column of Y and is the left eigenvector corresponding to eigenvalue λ_i. Therefore,

$$\boldsymbol{\eta}_i \in \mathcal{N}_i = \mathcal{N}(Q_c^T (A^T - \lambda_i I)), \tag{3.64}$$

where $\mathcal{N}$ represents the right null space. This gives some restriction on the choice of each column $\boldsymbol{\eta}_i$ of Y, but since $\mathcal{N}_i$ has dimension p (by observability, [47]), there

is still some freedom to choose the $\boldsymbol{\eta}_i$ if $p > 1$. We can use this freedom to ensure that our selected eigenvalues are as insensitive as possible to perturbations in A, C and G and thus that the system is *robust* [47]. This can help to reduce the effects of model error [79].

Eigenstructure assignment for robustness

The sensitivity of eigenvalue λ_i to perturbations in the components of A, C and G is given by

$$c_i = \frac{\|\boldsymbol{\xi}_i\|\|\boldsymbol{\eta}_i\|}{|\boldsymbol{\eta}_i^T \boldsymbol{\xi}_i|}, \tag{3.65}$$

where $\boldsymbol{\xi}_i$ are the columns of X, and $\boldsymbol{\eta}_i^T$ the rows of Y^T [88].

We note that

$$c_i = \frac{1}{|\cos \alpha|} \tag{3.66}$$

where α is the angle between $\boldsymbol{\eta}_i$ and $\boldsymbol{\xi}_i$, by the scalar product rule. Therefore c_i is smallest where α is smallest, which will be where $\boldsymbol{\eta}_i$ is parallel to $\boldsymbol{\xi}_i$. To optimize the conditioning, then, we choose each $\boldsymbol{\eta}_i$ to be as close as possible to parallel to $\boldsymbol{\xi}_i$. We have that $\boldsymbol{\xi}_i$ is orthogonal to $\boldsymbol{\eta}_j$ for all $j \neq i$ [39]. If $\boldsymbol{\eta}_i$ is to be parallel to $\boldsymbol{\xi}_i$, it follows that $\boldsymbol{\eta}_i$ should also be orthogonal to $\boldsymbol{\eta}_j$ for all $j \neq i$. A necessary condition for optimal conditioning is therefore that the vectors $\boldsymbol{\eta}_j$ be as close to orthogonal to each other as possible.

To summarize, our aim is to choose a set of vectors $\boldsymbol{\eta}_i$, the columns of Y so that for all $i = 1, 2, ..n$

a) $\boldsymbol{\eta}_i \in \mathcal{N}_i = \mathcal{N}(Q_c^T(A^T - \lambda_i)) \quad \forall i = 1, .., n$

b) the $\boldsymbol{\eta}_i$ are linearly independent

c) the $\boldsymbol{\eta}_i$ are as close to orthogonal to each other as possible.

Condition **b)** is included, because the inverse of Y is needed for evaluating G. The set of vectors $\boldsymbol{\eta}_i$ must be scaled so that $\|\boldsymbol{\eta}_i\| = 1$.

A method for eigenstructure assignment

The method described here involves choosing a set of vectors $\boldsymbol{\eta}_i$ which satisfy conditions **a**), **b**) and **c**) above, and follows a method given in [79].

Calculating the QR decomposition of $(A-\lambda_i I)Q_c$ gives

$$(A-\lambda_i I)Q_c = [\tilde{S}_i, S_i]\begin{bmatrix} R_i \\ 0 \end{bmatrix}, \tag{3.67}$$

where $[\tilde{S}_i, S_i]$ is orthogonal, $\tilde{S}_i$ is $n \times (n-p)$, $\quad S_i$ is $n \times p$, and R_i is $(n-p)\times(n-p)$ upper triangular and nonsingular. It follows by orthogonality of $[\tilde{S}_i, S_i]$, that

$$Q_c^T(A^T-\lambda_i I)S_i = 0. \tag{3.68}$$

Therefore, if $\boldsymbol{\eta}_i$ is in the space spanned by the columns of S_i, then condition **a**) is satisfied.

We now choose any set of linearly independent left eigenvectors $\boldsymbol{\eta}_i$ satisfying condition **a**), and modify these in turn to satisfy condition **c**). Let

$$Y_{-i} = \{\boldsymbol{\eta}_1, ..., \boldsymbol{\eta}_{i-1}, \boldsymbol{\eta}_{i+1}, ..., \boldsymbol{\eta}_n\}. \tag{3.69}$$

We want $\boldsymbol{\eta}_i$ to be as close to orthogonal as possible to this set. Calculating the QR decomposition gives

$$Y_{-i} = [\tilde{Z}_i, \mathbf{z}_i]\begin{bmatrix} \tilde{Y}_i \\ 0 \end{bmatrix}, \tag{3.70}$$

where $[\tilde{Z}_i, \mathbf{z}_i]$ is orthogonal, $\tilde{Y}_i$ is upper triangular and nonsingular, and $\mathbf{z}_i$ is an $n \times 1$ vector. This gives us the vector $\mathbf{z}_i$ which is orthogonal to Y_{-i}, but $\mathbf{z}_i$ may not be in $\mathcal{N}_i$, which would violate condition **a**). Choosing $\boldsymbol{\eta}_i$ to be the orthogonal projection of $\mathbf{z}_i$ into $\mathcal{N}_i$ ensures that $\boldsymbol{\eta}_i$ is as orthogonal as possible to the set Y_{-i} whilst satisfying condition **a**). So, after normalization we take

$$\boldsymbol{\eta}_i = S_i S_i^T \mathbf{z}_i / \|S_i S_i^T \mathbf{z}_i\|. \tag{3.71}$$

When all the columns have been modified in this way, the same procedure can then be repeated to modify the $\boldsymbol{\eta}_i$ again, until $\|(Y^T)^{-1}\|_F \equiv \sum_i c_i$ reaches a local minimum. Minimizing $\|(Y^T)^{-1}\|_F$ is a way of minimizing all the c_i together. The feedback matrix G can then be calculated from (3.61), using the Y derived.

This method for improving the robustness of the system can not be guaranteed to converge to the minimum possible value of $\|(Y^T)^{-1}\|_F$, but in practice it has been found to reduce its value significantly.

An algorithm for a robust observer

1) Calculate the QR decomposition of C^T into

$$C^T = \left[\tilde{Q}_c, Q_c\right] \begin{bmatrix} R_o \\ 0 \end{bmatrix}. \tag{3.72}$$

2) For each $i = 1, .., n$,
calculate the QR decomposition of $(A - \lambda_i I)Q_c$ into

$$(A - \lambda_i I)Q_c = \left[\tilde{S}_i, S_i\right] \begin{bmatrix} R_i \\ 0 \end{bmatrix}. \tag{3.73}$$

3) Choose columns from each of the S_i to be columns of the first guess Y, in such a way that Y is invertible.

4) For $i = 1, ..., n$, modify the columns $\boldsymbol{\eta}_i$ of Y as follows:

4a) calculate the QR decomposition of $Y_{-i} = \{\boldsymbol{\eta}_1, .., \boldsymbol{\eta}_{i-1}, \boldsymbol{\eta}_{i+1}, .., \boldsymbol{\eta}_n\}$ into

$$Y_{-i} = [\tilde{Z}_i, \mathbf{z}_i] \begin{bmatrix} \tilde{Y}_i \\ 0 \end{bmatrix}. \tag{3.74}$$

4b) project the vector $\mathbf{z}_i$ into space S_i to satisfy condition **a**) and then normalize:

$$\boldsymbol{\eta}_i = S_i S_i^T \mathbf{z}_i / \|S_i S_i^T \mathbf{z}_i\|. \tag{3.75}$$

5) Repeat Step 4 until $\|(Y^T)^{-1}\|_F$ reaches a local minimum.

6) Using the Y found, let the feedback matrix be G where

$$G^T = R_o^{-1} \tilde{Q}_c^T (A^T Y - Y D) Y^{-1}. \tag{3.76}$$

3.4 Extending 3D schemes to 4D

3.4.1 Introduction

3D data assimilation schemes are designed for an analysis at a single time. This approach was suitable in the earlier days of data assimilation, when observations were available mostly at synoptic times. For many modern applications of data assimilation, however, observations are available much more frequently.

The OI, 3DVAR and PSAS methods all have theoretical extensions to 4D schemes. For example, the Kalman gain matrix K_k given in (3.23) is the same as the OI weighting matrix W_k given in (3.7) with the matrix P_k of the OI scheme playing the rôle of the forecast error covariance matrix P_k^f of the Kalman filter. If the matrix P_k of the OI scheme is updated from P_{k-1} in the same way as the forecast error covariance matrices of the Kalman filter, then the OI scheme extended to 4D is equivalent to the Kalman filter. Since updating the forecast error covariance matrices is a very expensive part of Kalman filtering when the dimension of the system is large, in OI applications the covariance matrices P_k are usually kept constant, or a much simpler updating is performed, [32].

Similarly, the 3DVAR method can be extended to the strong constraint 4D variational assimilation method by summing the observational part of the cost function $\mathcal{J}$ given in (3.5) over the times that observations are available, and performing the minimization subject to the constraint that the model equations hold. In this method, it is only necessary to specify the covariance matrix P_0 at time t_0, as the matrices P_k do not need to be calculated explicitly.

These 4D data assimilation methods are much more expensive than the 3D methods, however. Although much recent research in data assimilation has centred around the theory and development of these 4D schemes, it is likely that 3D schemes will be used operationally at some centres for a while yet. There is also still interest in simple schemes, such as the successive correction method [38]. Successive correction methods are largely empirically designed, to "smooth in" observations to a prior estimate of the state at a single analysis time. In the next subsection, we discuss how ideas from control theory on observer design can be applied to provide

a theoretical extension of successive correction schemes to 4D. This could provide a way to make successive correction schemes more appropriate for use in a modern application of data assimilation in which observations are available more frequently.

3.4.2 Successive correction schemes as observers

Here we suppose that a successive correction method is to be used for data assimilation with observations available frequently, and we show how it may be regarded as an observer. If an observer is applied over an assimilation interval, then the analysed solution over that interval does not satisfy the model dynamics, but the observer dynamics. Hence, considering a sequential data assimilation method as an observer gives a different way of understanding some of the properties of the analysed solution.

In particular, by considering a successive correction scheme using observer theory, we are able to consider theoretical convergence *in time* of the scheme to the true model solution. In the data assimilation literature, the issue of whether a successive correction scheme *converges* generally refers to the question of whether the successive iterations or corrections (at a single analysis time) bring the analysis close to the true solution at that time [24]. In general, however, observations from more than one time are needed to determine the true state uniquely. Here, we consider whether the successive correction technique converges in time to the true model state.

We suppose that the evolution of the true model state is given by the linear, time invariant system

$$\mathbf{x}_{k+1}^t = A\mathbf{x}_k^t + B\mathbf{u}_k, \qquad k = 0, .., N-1 \tag{3.77}$$

as defined in (2.17) and we suppose that we have observations available at every timestep, related to the true model state by

$$\mathbf{y}_k = C\mathbf{x}_k^t, \qquad k = 0, .., N-1 \tag{3.78}$$

as defined in (2.18).

The successive correction method, with a constant number s of corrections, finds an analysis $\mathbf{x}_k^a$ from a prior estimate $\mathbf{x}_k^b$ using the following iteration

$$\mathbf{x}_k^{(i+1)} = \mathbf{x}_k^{(i)} + W^{(i+1)}(\mathbf{y}_k - C\mathbf{x}_k^{(i)}), \qquad i = 0, .., s-1, \tag{3.79}$$

with $\mathbf{x}_k^{(0)} = \mathbf{x}_k^b$, and $\mathbf{x}_k^a = \mathbf{x}_k^{(s)}$, and where the $W^{(i)}$, $i = 1, .., s$ represent the weighting matrices used in the successive corrections. After manipulation, the method can be written for theoretical purposes in the form

$$\mathbf{x}_k^a = \mathbf{x}_k^b + \tilde{W}^{(s)}(\mathbf{y}_k - C\mathbf{x}_k^b), \tag{3.80}$$

where the matrix $\tilde{W}^{(s)}$ is given by the recursion

$$\tilde{W}^{(i+1)} = W^{(i+1)}(I - C\tilde{W}^{(i)}) + \tilde{W}^{(i)}, \qquad i = 1, .., s-1, \tag{3.81}$$

with $\tilde{W}^{(1)} = W^{(1)}$.

From an analysis $\mathbf{x}_k^a$ at time t_k, the prior estimate for the next timestep, $\mathbf{x}_{k+1}^b$, is found using the model equations

$$\mathbf{x}_{k+1}^b = A\mathbf{x}_k^a + B\mathbf{u}_k, \qquad k = 0, .., N-1. \tag{3.82}$$

Substituting (3.82) into (3.80) gives

$$\mathbf{x}_{k+1}^a = A\mathbf{x}_k^a + B\mathbf{u}_k + \tilde{W}^{(s)}(\mathbf{y}_{k+1} - C\mathbf{x}_{k+1}^b), \tag{3.83}$$

which expresses the successive correction method in the form of a dynamic observer. Subtracting (3.77) from (3.83) and using (3.78) gives the following equation for the evolution of the error $\mathbf{e}_k = \mathbf{x}_k^a - \mathbf{x}_k^t$,

$$\mathbf{e}_{k+1} = A\mathbf{e}_k + \tilde{W}^{(s)}(C(A\mathbf{x}_k^t + B\mathbf{u}_k) - C(A\mathbf{x}_k^a + B\mathbf{u}_k)), \tag{3.84}$$

or

$$\mathbf{e}_{k+1} = (A - \tilde{W}^{(s)}CA)\mathbf{e}_k. \tag{3.85}$$

Hence, the successive correction scheme converges to the true model state $\mathbf{x}_k^t$ in time if the eigenvalues of $(A - \tilde{W}^{(s)}CA)$ have modulus less than unity. The weighting matrix $\tilde{W}^{(s)}$ plays a similar rôle as the feedback matrix G in the observer (3.51). The matrix C in (3.52) has been replaced by the matrix product CA in (3.85), because (3.83) uses observations at time t_{k+1}, rather than at time t_k as the observer (3.51) does.

If observations are available less frequently, then $\mathbf{x}_{k+1}^a$ is specified by (3.83) when observations are available, and is equal to $\mathbf{x}_{k+1}^b$ given in (3.82) when observations

are not available. If observations are available every r^{th} timestep, then the error $\mathbf{e}_k$ satisfies

$$\mathbf{e}_{r(k+1)} = (A - \tilde{W}^{(s)}CA)A^{r-1}\mathbf{e}_{rk}, \tag{3.86}$$

and hence the successive correction scheme converges to the true model sate in time if the eigenvalues of $(A - \tilde{W}^{(s)}CA)A^{r-1}$ have modulus less than unity.

Discussion

In Section 3.3 on dynamic observer theory we discussed how the feedback matrix G could be designed to ensure convergence in time to the true solution, and so that the observer system has desirable properties (we considered good convergence and robustness). This theory could be used to provide a choice of the weighting matrices in the successive correction scheme which give it desirable dynamical behaviour. In Section 3.5, we illustrate this point with a simple example in which the robust observer described in Section 3.3 produces much better convergence in data sparse areas than the Cressman successive correction scheme.

3.5 An example comparing the Cressman scheme and robust observer

In this section, we compare the Cressman scheme, an example of a successive correction data assimilation scheme, with the robust observer described in Section 3.3, which is designed for good convergence and robustness. We first describe the simple model we use, and the observations we suppose are available. We then describe the experiments that are carried out, and discuss the results.

3.5.1 The models and observations

The model we use here is also used in the experiments of Chapter 5, and so we describe it in some detail.

The theta method for the 1D heat equation

The 1D heat equation on $z \in [0,1]$, $t \in [0,T]$, with a point heat source of strength $\frac{1}{3}$ at $z = \frac{1}{4}$ is,

$$v_t = \sigma v_{zz} + \frac{1}{3}\delta(z - \frac{1}{4}), \tag{3.87}$$

where δ is the Dirac delta function. For this equation, with initial condition

$$v(z,0) = \alpha(z), \tag{3.88}$$

and zero boundary conditions

$$v(0,t) = 0, \quad v(1,t) = 0, \tag{3.89}$$

the "theta method" discretisation for some $\theta \in [0,1]$ is

$$x_j^{k+1} - x_j^k = \frac{\sigma \Delta t}{\Delta z^2}\left\{(1-\theta)\delta^2 x_j^k + \theta \delta^2 x_j^{k+1}\right\} + s_j \Delta t \tag{3.90}$$

with initial condition

$$x_j^0 = \alpha(j\Delta z), \tag{3.91}$$

and zero boundary conditions

$$x_0^k = 0, \quad x_J^k = 0, \tag{3.92}$$

where $x_j^k \approx v(j\Delta z, k\Delta t)$ for $j = 0,1,..,J$, $k = 0,1,..,N$ with $\Delta z = \frac{1}{J}$ and $\Delta t = \frac{T}{N}$, and where $\delta^2 x_j^k$ denotes $x_{j-1}^k - 2x_j^k + x_{j+1}^k$. The dimension of this system is n, where $n = J - 1$. Here, we consider only the explicit form of (3.90), so we take $\theta = 0$.

As discussed in [39], the source term

$$s(z) = \frac{1}{3}\delta(z - \frac{1}{4}) \tag{3.93}$$

can be represented in discrete form with $s_j \approx s(j\Delta z)$ given by

$$s_j = \begin{cases} \frac{1}{3\Delta z} & \text{if} \quad j = \frac{J}{4} \\ 0 & \text{otherwise.} \end{cases} \tag{3.94}$$

The discretisation (3.90) (with $\theta = 0$) can be written as the matrix system,

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{s}, \tag{3.95}$$

where the state $\mathbf{x}_k$ at time t_k is given by

$$\mathbf{x}_k = (x_1^k, x_2^k, ..., x_n^k)^T. \tag{3.96}$$

The input $\mathbf{s} \in \mathbb{R}^n$ represents the source term and zero boundary conditions and is given by

$$\mathbf{s} = (0, ..., \frac{\Delta t}{3\Delta z}, ..., 0)^T, \tag{3.97}$$

where the non-zero element of $\mathbf{s}$ is s_j where $j = J/4$. The matrix $A \in \mathbb{R}^{n \times n}$ is given by

$$A = \begin{pmatrix} (1-2\mu) & \mu & & \\ \mu & (1-2\mu) & \mu & \\ \ddots & \ddots & \ddots & \end{pmatrix}, \tag{3.98}$$

where $\mu = \sigma \frac{\Delta t}{\Delta z^2}$. The theta method with $\theta = 0$ is stable for $0 \leq \mu \leq \frac{1}{2}$.

Observational data

We suppose that we have p observations ($1 \leq p \leq n$) at each of N timesteps, given by

$$\mathbf{y}_k = C\mathbf{x}_k^t, \qquad k = 0, .., N-1. \tag{3.99}$$

The matrix C represents a linear interpolation between the model grid and the p observation positions on the interval $[0, 1]$, specified in Table 3.1 below.

Table 3.1: The observation positions

obs_1	obs_2	obs_3	obs_4	obs_5	obs_6	obs_7	obs_8	...
0.03	0.12	0.19	0.26	0.37	0.42	0.45	0.56	...

...	obs_9	obs_{10}	obs_{11}	obs_{12}	obs_{13}	obs_{14}	obs_{15}
...	0.57	0.60	0.67	0.71	0.73	0.83	0.92

The matrix C is built up as follows: if observation i (where $1 \leq i \leq p$) has the position obs_i which lies between grid points j and $j+1$, then

$$\begin{aligned} C_{i,j} &= \frac{(j+1)\Delta z - obs_i}{\Delta z}, \\ C_{i,j+1} &= \frac{obs_i - j\Delta z}{\Delta z}, \\ C_{i,k} &= 0 \qquad k \neq j, \quad k \neq j+1. \end{aligned} \tag{3.100}$$

If obs_i lies between either the first or n^{th} grid point and its adjacent boundary point, then row i of C has just one non-zero entry, since the boundary conditions are zero.

For example, with $p = 5$, C is the $5 \times n$ matrix

$$C = \begin{pmatrix} 0.48 & 0 & 0 & ... & & & & \\ 0.08 & 0.92 & 0 & 0 & ... & & & \\ 0 & 0 & 0.96 & 0.04 & 0 & ... & & \\ 0 & 0 & 0 & 0.84 & 0.16 & 0 & ... & \\ 0 & 0 & 0 & 0 & 0.08 & 0.92 & 0 & ... \end{pmatrix}. \tag{3.101}$$

3.5.2 Description of the experiments

We suppose that the evolution of the true model state $\mathbf{x}_k^t$ is given by the model

$$\mathbf{x}_{k+1}^t = A\mathbf{x}_k^t + \mathbf{s}, \qquad k = 0, .., N-1 \tag{3.102}$$

as defined in (3.95), with initial conditions

$$(x^t)_j^0 = 1, \qquad j = 1, .., n. \tag{3.103}$$

We set $N = 40$ and $T = \frac{1}{2}$, (hence $\Delta t = \frac{1}{80}$), and $J = 16$ (hence $\Delta z = \frac{1}{16}$, and $n = 15$) and $\sigma = 0.1$ (hence $\mu = 0.32$).

We suppose that we have p (error free) observations at N timesteps, and that these are related to the true model state by

$$\mathbf{y}_k = C\mathbf{x}_k^t, \qquad k = 0, .., N-1, \tag{3.104}$$

as defined in (3.99). We suppose that the true initial state (3.103) is unknown, and that our "prior estimate" of the initial state, $\mathbf{x}_0^b$, is given by

$$(x^b)_j^0 = 2, \qquad j = 1, .., n. \tag{3.105}$$

In these experiments, we compare the Cressman scheme, a successive correction method for data assimilation which we described in Section 3.1.1, with the robust observer we described in Section 3.3.3, for different values of p, the number of observations. The experiments carried out are as follows.

Data assimilation using the Cressman scheme

Since this is a simple example, the Cressman scheme is implemented using only one correction ($s = 1$) using just one radius of influence R. The experiments are carried out using different values for R: $R = 0.1$, $R = 0.3$, $R = 0.5$ and $R = 0.9$, and different values for p, the number of observations available at each timestep.

Data assimilation using the robust observer

We let Λ_m denote the eigenvalues of the model (3.95). In this experiment, different sets of eigenvalues are assigned to the observer: Λ_a, Λ_b and Λ_c. The set of eigenvalues Λ_a are equally distributed between -0.5 and 0.5. The sets of eigenvalues Λ_b and Λ_c represent the model eigenvalues reduced in modulus by a quarter and by a half, respectively, ie the eigenvalues in the set Λ_m multiplied by 0.75 and 0.5, respectively.

3.5.3 Results

The figures referred to here can be found at the end of this section.

The Cressman scheme

Figure 3.1 shows that when a large number of observations are used ($p = 10$), complete convergence to the true solution is achieved in approximately 40 timesteps, using $R = 0.3$ as the radius of influence. When R is reduced to 0.1, convergence takes about 60 timesteps.

When fewer observations are used, convergence to the true solution occurs quickly in data dense areas, but much more slowly in data sparse areas. Fig. 3.2 illustrates this for the case $p = 5$, $R = 0.3$. In this case, the solution with data assimilation is closer to the true solution in the data sparse areas than the solution without data

assimilation is, but it does not have the spatial shape of the true solution. Fig. 3.3 shows the case $p = 1$, where only one observation is available right near one of the boundaries of the domain, using $R = 0.3$. In this case, the data assimilation has only a small impact on the results. Increasing R to 0.9, so that the radius of influence extends all the way across the spatial domain, convergence is still slow, as Fig. 3.4 shows.

The robust observer

Very good results are achieved using eigenvalue set Λ_b, in which we assign to the observer system (ie, to the matrix $(A - CG)$) the system eigenvalues multiplied by 0.75. In this case convergence to the true solution is achieved in fewer than 20 timesteps using 5 observations, as Fig. 3.5 shows. Using eigenvalue set Λ_a, in which the eigenvalues to be assigned are evenly distributed between -0.5 and 0.5 gives less pleasing results. From this it seems that it is important for good convergence to reduce the modulus of all the eigenvalues, as we do when assigning eigenvalue set Λ_b, but not when assigning set Λ_a. We give more detail on experiments in assigning different sets of eigenvalues in a report on observers and data assimilation, [39].

One pleasing aspect of these results compared with those obtained using the Cressman scheme is that there is fast convergence in data sparse areas. Figure 3.5 illustrates this in the case $p = 5$, and Fig. 3.6 in the case $p = 1$, where eigenvalue set Λ_b is used. Using eigenvalue set Λ_c (so the eigenvalues are 0.5 times the size of the system eigenvalues) gives slightly faster convergence in the cases where few observations are used, and complete convergence is achieved in less than 30 timesteps using only one observation, as Fig. 3.7 shows.

Discussion

The model used in these esperiments is not really ideal for testing, since solutions converge quickly to a steady state. Even so, these simple experiments illustrate some interesting points.

Although the Cressman scheme gives good convergence near the observation positions, convergence is slow in data sparse areas. The robust observer, however,

produces much faster convergence in data sparse areas. This serves as an example of how designing the feedback matrix of an observer to ensure temporal convergence to the true solution can improve on the empirical spatial smoothing of a successive correction method. The robust observer design itself, involving eigenstructure assignment, however, would be too expensive for systems with very large dimension, and hence for application to operational data assimilation.

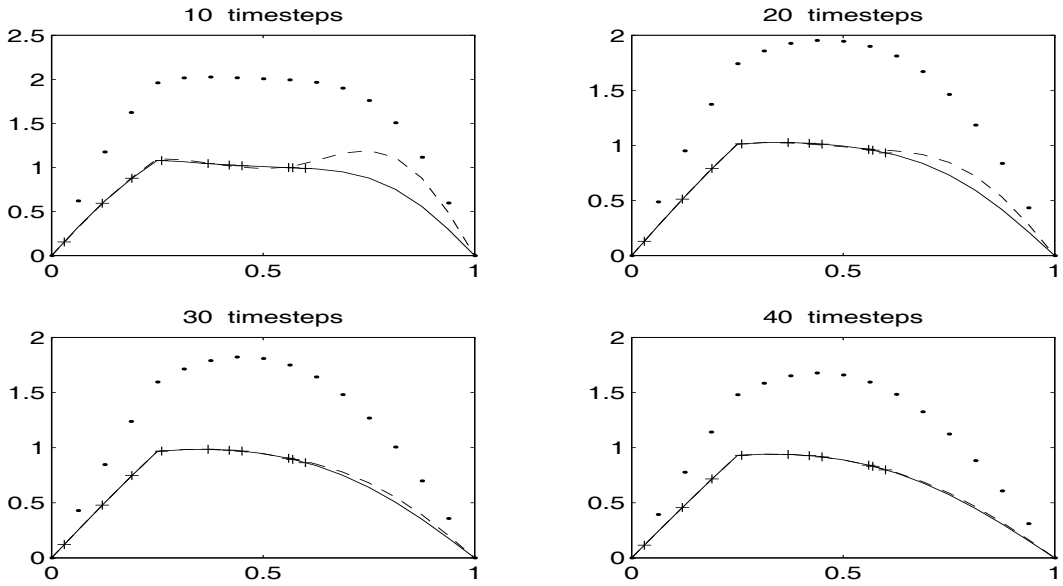

Figure 3.1: Data assimilation using the Cressman scheme with $R = 0.3$ using 10 observations. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

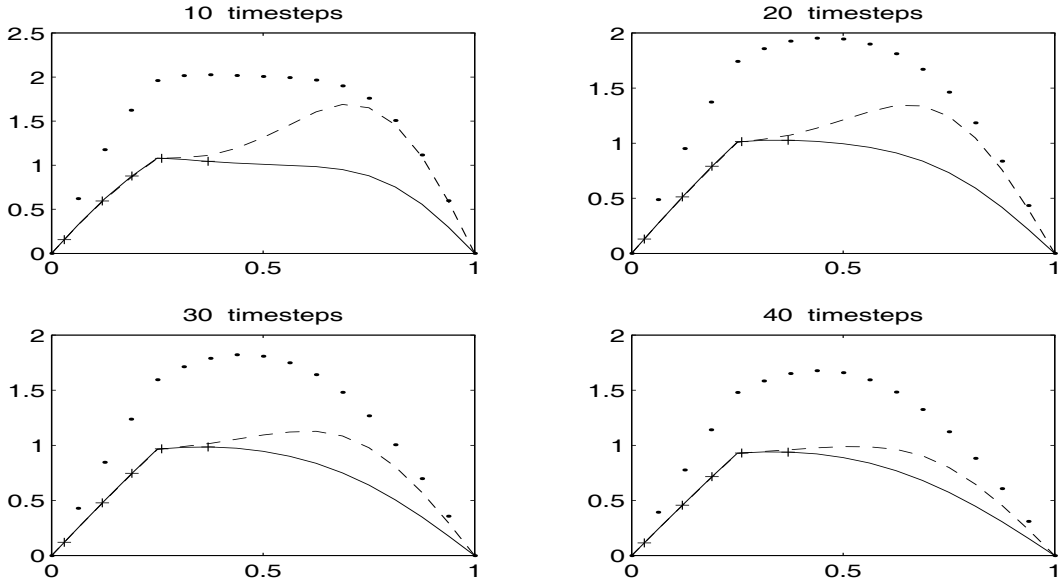

Figure 3.2: Data assimilation using the Cressman scheme with $R = 0.3$ using 5 observations. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

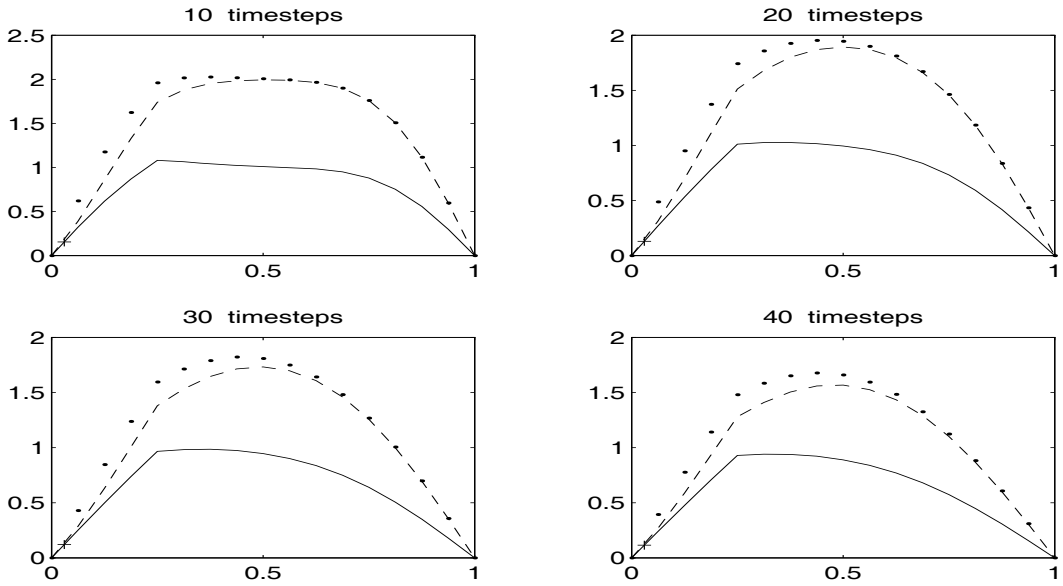

Figure 3.3: Data assimilation using the Cressman scheme with $R = 0.3$ using 1 observation. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

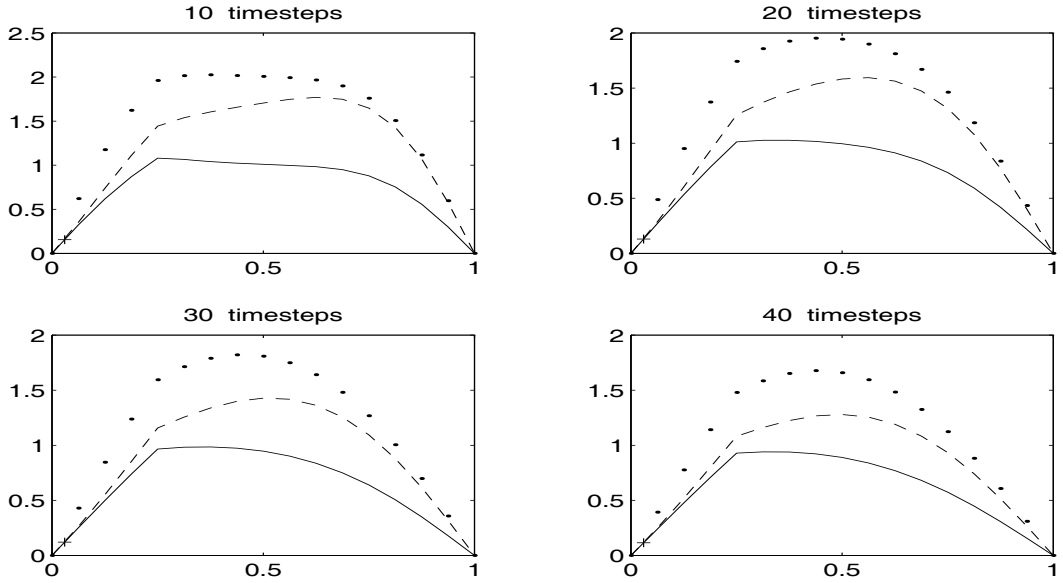

Figure 3.4: Data assimilation using the Cressman scheme with $R = 0.9$ using 1 observation. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

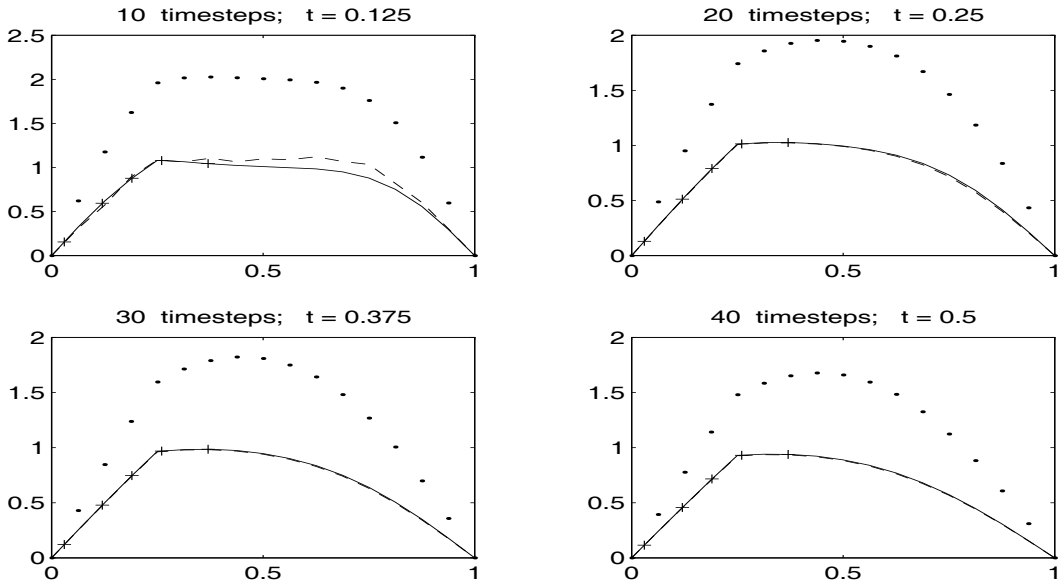

Figure 3.5: Data assimilation using the robust observer with eigenvalue set Λ_b using 5 observations. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

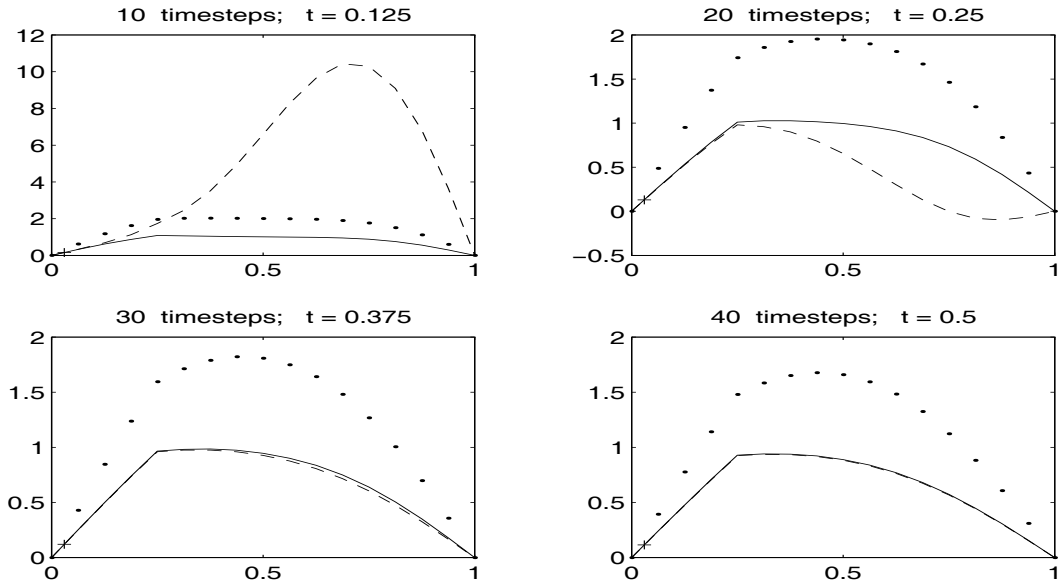

Figure 3.6: Data assimilation using the robust observer with eigenvalue set Λ_b using 1 observation. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

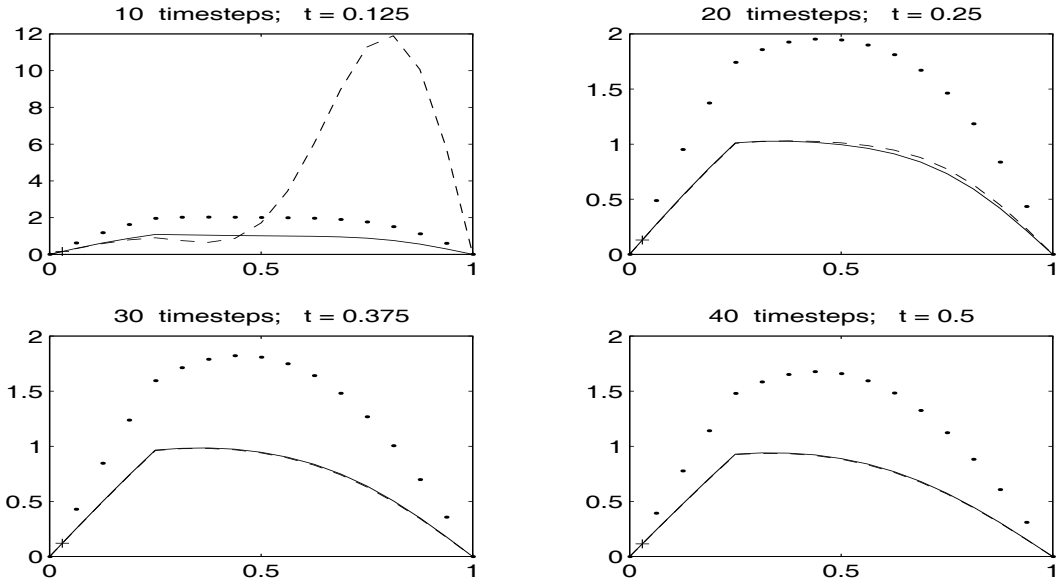

Figure 3.7: Data assimilation using the robust observer with eigenvalue set Λ_c using 1 observation. Solid line: true solution; dotted line: solution with no assimilation; dashed line: solution with assimilation, crosses: observations.

Chapter 4

4D Variational assimilation

4D variational methods of data assimilation were introduced to meteorology by Sasaki in his paper of 1958 [75]. These schemes seek to find the model state which minimizes some cost function over a particular assimilation interval, subject to constraints on the model state. Most typically the constraints require that the model state should satisfy the dynamical model equations over the assimilation time period.

Sasaki put forward two approaches to variational assimilation. In the *strong constraint* approach, the solution is constrained to satisfy the model equations exactly. In the *weak constraint* approach, the model equations are required to hold only approximately, allowing for model error. Sasaki's papers [75], [76], [77], [78], deal with methods of solving these minimization problems analytically for simple, continuous models.

Various methods for solving the strong constraint problem are outlined in [32]. One method is to iterate on the model initial state rather than on the model state over the whole assimilation interval. This technique of "reducing the control vector" which we outlined in Chapter 2, Section 2.3, significantly reduces the cost of variational assimilation. In this case the initial state is the control vector. The method was introduced to meteorology in the mid 1980s in the papers by Le Dimet and Talagrand [51], Lewis and Derber [52], Lorenc [55], and Courtier and Talagrand [18], [80]; and to oceanography by Thacker and Long [83]. It is currently being developed for implementation as an operational data assimilation scheme at several national meteorological centres [73].

Derber [26] suggested carrying out 4D variational assimilation, adding to the model equations a correction term which is constant over the assimilation interval, and which approximates model error. This "correction term technique" is a modification of the strong constraint method, in which the correction term is used instead of, or as well as, the initial state as a control vector. The weak constraint approach, which allows for model error without the approximation that model error is constant over the assimilation interval, is a more difficult problem, however.

This chapter is organised as follows. In Section 4.1, we introduce the strong constraint problem and describe the technique of reducing the control vector for solving this problem. In Section 4.2, we give a discussion on the development of the *adjoint models* which form a central part of the method. Then in Section 4.3 we outline Derber's correction term technique, and finally in Section 4.4 we describe the weak constraint approach, which allows for model error. In this context we also discuss the links between the 4D variational methods and other methods of data assimilation, and state conditions under which the solution is statistically optimal.

Throughout the chapter, we consider the nonlinear model system given by

$$\mathbf{x}_{k+1}^t = \mathbf{f}_k(\mathbf{x}_k^t) + \boldsymbol{\varepsilon}_k, \qquad k = 0, .., N-1, \tag{4.1}$$

as defined in (2.4), where $\mathbf{x}_k^t \in \mathbb{R}^n$ is the true model state at time t_k, $\mathbf{f}_k : \mathbb{R}^n \to \mathbb{R}^n$ represents the nonlinear evolution of the state from time t_k to t_{k+1}, and $\boldsymbol{\varepsilon}_k \in \mathbb{R}^n$ represents model error. Here, for notational convenience, we do not indicate dependence on the model inputs $\mathbf{u}_k$, which we suppose are fixed. We suppose we have observations $\mathbf{y}_k \in \mathbb{R}^{p_k}$ at time t_k, related to the true model state by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1, \tag{4.2}$$

as defined in (2.6), where $\mathbf{h}_k : \mathbb{R}^n \to \mathbb{R}^{p_k}$ is a nonlinear operator, and $\boldsymbol{\delta}_k \in \mathbb{R}^{p_k}$ is the observational error. Finally, we suppose that we have a prior estimate $\mathbf{x}_0^b \in \mathbb{R}^n$ of the initial state, called a *background estimate* in the context of variational assimilation, satisfying

$$\mathbf{x}_0 = \mathbf{x}_0^b + \boldsymbol{\beta}_0, \tag{4.3}$$

where $\boldsymbol{\beta}_0 \in \mathbb{R}^n$ is the *background error*. At this stage, we do not make any assumptions about the statistics of the errors $\boldsymbol{\varepsilon}_k$, $\boldsymbol{\delta}_k$ and $\boldsymbol{\beta}_0$.

4.1 The strong constraint approach

In the strong constraint approach, model error is neglected and we work with the model

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k), \qquad k = 0, .., N-1. \tag{4.4}$$

We suppose that we have observations given by (4.2) and a background estimate of the initial state given by (4.3).

4.1.1 The method

In the strong constraint approach to variational assimilation, we aim to minimize a cost function $\mathcal{J}$ with three components,

$$\mathcal{J} = \mathcal{J}_b + \mathcal{J}_o + \mathcal{J}_c, \tag{4.5}$$

where $\mathcal{J}_b$ penalizes distance from the background estimate, $\mathcal{J}_o$ penalizes distance from the observations, and $\mathcal{J}_c$ ensures that the solution has required smoothness properties, so that the process of initialization (or part of it) mentioned in Chapter 1 can be incorporated in the optimization procedure. The work in this thesis does not include the component $\mathcal{J}_c$ for simplicity, although it is important in operational applications of data assimilation [91], [94], [19], that use more complex models and fewer observations than we use in our idealized experiments.

The strong constraint problem we address is

Problem $\mathcal{S}$

Minimize, with respect to $\mathbf{x}_0, .., \mathbf{x}_N$

$$\mathcal{J} = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^T P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j)^T R_j^{-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j), \tag{4.6}$$

subject to (4.4)

The matrices $P_0^{-1} \in \mathbb{R}^{n\times n}$ and $R_j^{-1} \in \mathbb{R}^{p_j \times p_j}$ are symmetric positive definite weighting matrices which reflect the accuracies of $(\mathbf{x}_0 - \mathbf{x}_0^b)$ and of $(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j)$. If the inverse covariance matrices of the errors $\boldsymbol{\beta}_0$ and $\boldsymbol{\delta}_j$ are known and are nonsingular, these can be used as weighting matrices, and under certain assumptions this choice

leads to a statistically optimal analysis. We discuss this further in Section 4.4 where we also consider model error. Some detail on how the matrices P_0^{-1} and R_j^{-1} are prescribed in practice is given in [13], [57].

We now use the theory of Chapter 2, Section 2.3, on reducing the control vector to give a method for solving the constrained minimization problem Problem $\mathcal{S}$. We have the $N+1$ state vectors $\mathbf{x}_0, .., \mathbf{x}_N$ as the unknown variables, and (4.4) specifies N constraints on these. If we use $\mathbf{x}_0$ the control vector, we can determine the remaining N state vectors $\mathbf{x}_1, .., \mathbf{x}_N$ uniquely from the N constraints (4.4).

The constrained minimization problem Problem $\mathcal{S}$ is equivalent to the unconstrained optimization problem of extremizing the Lagrangian function

$$\mathcal{L} = \mathcal{J} + \sum_{j=0}^{N-1} \boldsymbol{\lambda}_{j+1}^T (\mathbf{x}_{j+1} - \mathbf{f}_j(\mathbf{x}_j)), \tag{4.7}$$

with respect to $\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_N$, and $\boldsymbol{\lambda}_1, .., \boldsymbol{\lambda}_N$, where $\boldsymbol{\lambda}_1, .., \boldsymbol{\lambda}_N \in \mathbb{R}^n$ are N vectors of Lagrange multipliers. A necessary condition for an extremal is that the gradient of $\mathcal{L}$ with respect to $\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_N$, and $\boldsymbol{\lambda}_1, .., \boldsymbol{\lambda}_N$ should vanish. Using the method of reducing the control vectors, this necessary condition can be achieved by iterating on the control vector $\mathbf{x}_0$, as follows. From a guess of the control vector, the model states $\mathbf{x}_1, .., \mathbf{x}_N$ are calculated using the constraints (4.4). This ensures that $\nabla_{\boldsymbol{\lambda}_k} \mathcal{L} = 0$ for $k = 1, .., N$. The adjoint vectors $\boldsymbol{\lambda}_k$ must now be chosen to ensure $\nabla_{\mathbf{x}_k} \mathcal{L} = 0$ for $k = 1, .., N$, ie,

$$0 = \nabla_{\mathbf{x}_k} \mathcal{J} + \boldsymbol{\lambda}_k - F_k^T(\mathbf{x}_k) \boldsymbol{\lambda}_{k+1}, \qquad k = 1, .., N-1, \tag{4.8}$$

$$0 = \boldsymbol{\lambda}_N, \tag{4.9}$$

where $F_k(\mathbf{x}_k) \in \mathbb{R}^{n \times n}$ is the Jacobian of $\mathbf{f}_k(\mathbf{x}_k)$ with respect to $\mathbf{x}_k$. Since

$$\nabla_{\mathbf{x}_k} \mathcal{J} = H_k^T(\mathbf{x}_k) R_k^{-1} (\mathbf{h}_k(\mathbf{x}_k) - \mathbf{y}_k), \tag{4.10}$$

where $H_k(\mathbf{x}_k) \in \mathbb{R}^{p_k \times n}$ is the Jacobian of $\mathbf{h}_k$ with respect to $\mathbf{x}_k$, the Lagrange multipliers must satisfy

$$\boldsymbol{\lambda}_k = F_k^T(\mathbf{x}_k) \boldsymbol{\lambda}_{k+1} - H_k^T(\mathbf{x}_k) R_k^{-1} (\mathbf{h}_k(\mathbf{x}_k) - \mathbf{y}_k), \qquad k = 1, .., N-1, \tag{4.11}$$

$$\boldsymbol{\lambda}_N = 0. \tag{4.12}$$

We note that, since $\mathbf{x}_1, .., \mathbf{x}_N$ have been calculated from the current guess of the control vector, the vectors $\boldsymbol{\lambda}_k$ of Lagrange multipliers can be calculated recursively backwards from the condition (4.12).

The system of equations (4.11), (4.12) is known as the system of *adjoint equations* for the model (4.4), or as the *adjoint model*. In this context, the Lagrange multipliers are known as *adjoint variables*, and we refer to the vectors $\boldsymbol{\lambda}_k$ of adjoint variables as *adjoint vectors*.

We can now evaluate the gradients of $\mathcal{L}$ with respect to the control vectors. The gradient with respect to the initial state is given by

$$\begin{aligned} \nabla_{\mathbf{x}_0}\mathcal{L} &= \nabla_{\mathbf{x}_0}\mathcal{J} - F_0^T(\mathbf{x}_0)\boldsymbol{\lambda}_1 && (4.13)\\ &= P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + H_0^T(\mathbf{x}_0)R_0^{-1}(\mathbf{h}_0(\mathbf{x}_0) - \mathbf{y}_0) - F_0^T(\mathbf{x}_0)\boldsymbol{\lambda}_1 && (4.14)\\ &= P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) - \boldsymbol{\lambda}_0, && (4.15) \end{aligned}$$

where the additional adjoint vector $\boldsymbol{\lambda}_0 \in \mathbb{R}^n$ is defined via the relation (4.11) with $k = 0$. This gradient can be used in a descent algorithm, such as one outlined in Chapter 2, Section 2.4, to improve our guess of the control vector. We summarize this procedure in the following algorithm.

Algorithm IS

1. From a guess of the control vector $\mathbf{x}_0$,
   calculate the model states $\mathbf{x}_1, .., \mathbf{x}_N$ using the model equations (4.4).

2. From the end condition (4.12), calculate the adjoint vectors $\boldsymbol{\lambda}_{N-1}, .., \boldsymbol{\lambda}_0$ using the model states calculated in Step 1.

3. From $\boldsymbol{\lambda}_0$, calculate $\nabla_{\mathbf{x}_0}\mathcal{L}$ using (4.15)

4. Use the gradient $\nabla_{\mathbf{x}_0}\mathcal{L}$ in a gradient algorithm to obtain a better guess of the control vector $\mathbf{x}_0$, and repeat until convergence criteria are satisfied.

4.1.2 The incremental approach for Problem $\mathcal{S}$

Although the method of reducing the control vector significantly reduces the expense of 4D variational assimilation, further reductions in its expense are still required to

make it feasible for operational implementation. The incremental approach [20] was suggested to allow flexibility to incorporate simplifications which will reduce the expense of the method. We first describe the incremental approach, and then describe the approximations that can be made to reduce expense.

Expanding the nonlinear model (4.4) in a Taylors series about the "background" state $\mathbf{x}_k^b$ obtained from a model run using the model with $\mathbf{x}_0^b$ as the initial state, we have for a small perturbation $\boldsymbol{\delta}\mathbf{x}_k$ of $\mathbf{x}_k$,

$$\mathbf{x}_{k+1}^b + \boldsymbol{\delta}\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k^b) + F_k(\mathbf{x}_k^b)\boldsymbol{\delta}\mathbf{x}_k + \mathbf{o}(\boldsymbol{\delta}\mathbf{x}_k), \qquad k = 0, .., N-1, \tag{4.16}$$

where $F_k(\mathbf{x}_k^b)$ is the Jacobian of $\mathbf{f}_k$ with respect to $\mathbf{x}_k$ evaluated at $\mathbf{x}_k^b$, and $\mathbf{o}(\boldsymbol{\delta}\mathbf{x}_k)$ represents the higher order terms in the expansion. Since (4.4) holds at $\mathbf{x}_k^b$, we have, after neglecting higher order terms,

$$\boldsymbol{\delta}\mathbf{x}_{k+1} = F_k(\mathbf{x}_k^b)\boldsymbol{\delta}\mathbf{x}_k, \tag{4.17}$$

which is referred to as the *tangent linear model* (TLM). For mid-latitude meteorological models, Lacarra and Talagrand [48] have shown that the the TLM is a fair approximation to the full nonlinear model for periods of up to around 48 hours.

The observations $\mathbf{y}_k$ are related to the perturbation $\boldsymbol{\delta}\mathbf{x}_k^t = (\mathbf{x}_k^t - \mathbf{x}_k^b)$ as follows

$$\begin{aligned} \mathbf{y}_k &= \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k = \mathbf{h}_k(\mathbf{x}_k^b + \boldsymbol{\delta}\mathbf{x}_k^t) + \boldsymbol{\delta}_k & (4.18)\\ &\approx \mathbf{h}_k(\mathbf{x}_k^b) + H_k(\mathbf{x}_k^b)\boldsymbol{\delta}\mathbf{x}_k^t + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1. & (4.19) \end{aligned}$$

The incremental approach to solving Problem $\mathcal{S}$ proceeds as follows. Firstly, a background run is performed from a background guess $\mathbf{x}_0^b$ of the initial state, using the full nonlinear model (4.4) to calculate the terms $\mathbf{x}_k^b$. The minimization problem is then to find the optimal *increment* or perturbation $\boldsymbol{\delta}\mathbf{x}_0$ to $\mathbf{x}_0^b$, by minimizing a cost function of the form

$$\mathcal{J}(\boldsymbol{\delta}\mathbf{x}_0) = \frac{1}{2}\boldsymbol{\delta}\mathbf{x}_0^T P_0^{-1}\boldsymbol{\delta}\mathbf{x}_0 + \frac{1}{2}\sum_{j=0}^{N-1}(H_j(\mathbf{x}_j^b)\boldsymbol{\delta}\mathbf{x}_j - \mathbf{d}_j)^T R_j^{-1}(H_j(\mathbf{x}_j^b)\boldsymbol{\delta}\mathbf{x}_j - \mathbf{d}_j) \tag{4.20}$$

subject to the constraints (4.17), $k = 0, .., N-1$, where

$$\mathbf{d}_k = \mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^b). \tag{4.21}$$

Since all the constraints (4.17) are linear, we now have a cost function which is quadratic in the control vector $\boldsymbol{\delta}\mathbf{x}_0$, and so a unique global minimum to this problem exists if the Hessian of $\mathcal{J}$ with respect to $\boldsymbol{\delta}\mathbf{x}_0$ (satisfying the constraints (4.17)) is positive definite. We give conditions for this to hold in Chapter 5, Section 5.2. The adjoint equations are, in this case,

$$\boldsymbol{\lambda}_k = F_k^T(\mathbf{x}_k^b)\boldsymbol{\lambda}_{k+1} - H_k^T(\mathbf{x}_k^b)R_k^{-1}(H_k(\mathbf{x}_k^b)\boldsymbol{\delta}\mathbf{x}_k - \mathbf{d}_k), \quad k = 0, .., N-1, \tag{4.22}$$

$$\boldsymbol{\lambda}_N = 0, \tag{4.23}$$

but using (4.21) and (4.19) we see these are the same as the adjoint equations (4.11) of the full, nonlinear system, except that the higher order terms of (4.19) have been neglected in the forcing in (4.22), and that the Jacobian F_k is evaluated at $\mathbf{x}_k^b$.

If $\boldsymbol{\delta}\mathbf{x}_0^*$ is the control vector which solves the incremental problem, then $\mathbf{x}_0^* := \mathbf{x}_0^b + \boldsymbol{\delta}\mathbf{x}_0^*$, (and hence the model states $\mathbf{x}_1^*, .., \mathbf{x}_N^*$ found from this initial state) is a good approximation to the solution of the full, nonlinear mimimization problem, provided the TLM is a "valid" approximation to the full nonlinear model.

The incremental approach can be used to further reduce the cost of 4D variational assimilation by performing the background run to calculate the $\mathbf{x}_k^b$ using the full nonlinear model (4.4), but carrying out the iteration on $\boldsymbol{\delta}\mathbf{x}_0$ at lower resolution. The iteration at lower resolution could also be performed using a simplified (and hence less expensive) version of the TLM. Research is also being carried out on variants of the incremental approach. These include applying the incremental approach at lower resolution, perhaps using multi-grid strategies (this is the so-called multi-incremental approach), and interspersing several "inner loop" iterations with an "outer loop" nonlinear run, in which a new background field for the next inner loop iterations is obtained. One question being looked at both theoretically and practically, is whether the low resolution inner loop iterations give improvements which correspond to an improvement at full resolution, and whether these methods do converge to a solution of the full nonlinear problem.

Several centres planning to implement the adjoint method for large models are developing simplified or modified tangent linear models for the minimization using the incremental approach. This gives a way of overcoming some of the problems of

the adjoint method, for example by ensuring that the modified linear model is fully differentiable. At the UK Meteorological Office, the linear version of the full model being developed for use in data assimilation is called the "perturbation forecast" model, since it is not actually tangent linear to the full model [49]. Other centres are developing tangent linear models with simplified physics as an intermediate step to developing a complete tangent linear model [94].

4.2 Development of adjoint models

4.2.1 Properties of adjoint models

Before proceeding with a discussion on the practical development of an adjoint model, we note a further theoretical property of the adjoint equations. We suppose $\Phi(k,j)$ is the state transition matrix associated with the (unforced) tangent linear model (4.17). (Chapter 2, Section 2.1 gives background on state transition matrices.) We have

$$\boldsymbol{\delta}\mathbf{x}_k = \Phi(k,j)\boldsymbol{\delta}\mathbf{x}_j, \qquad \text{for all} \quad k \geq j, \tag{4.24}$$

where

$$\Phi(k,j) = F_{k-1}(\mathbf{x}^b_{k-1})F_{k-2}(\mathbf{x}^b_{k-2})...F_j(\mathbf{x}^b_j). \tag{4.25}$$

If we let $\Psi(j,k)$ be the state transition matrix for the unforced version of adjoint system (4.22), ie for the system

$$\tilde{\boldsymbol{\lambda}}_k = F^T_k(\mathbf{x}^b_k)\tilde{\boldsymbol{\lambda}}_{k+1}, \qquad k = N-1,..,0 \tag{4.26}$$

with end condition (4.23), we have

$$\tilde{\boldsymbol{\lambda}}_j = \Psi(j,k)\tilde{\boldsymbol{\lambda}}_k, \qquad \text{for all} \quad k \geq j, \tag{4.27}$$

where

$$\Psi(j,k) = F^T_j(\mathbf{x}^b_j)F^T_{j+1}(\mathbf{x}^b_{j+1})...F^T_{k-1}(\mathbf{x}^b_{k-1}). \tag{4.28}$$

Hence, we have the following [2], [51],

$$\Psi(j,k) = \Phi^T(k,j) \qquad \text{for all} \quad k \geq j. \tag{4.29}$$

Since $\Phi^T(k,j)$ is the *adjoint* operator of $\Phi(k,j)$ with respect to the Euclidean inner product, we see one reason why the adjoint equations are so-called.

Hence we have the following property,

$$< \tilde{\boldsymbol{\lambda}}_k, \boldsymbol{\delta}\mathbf{x}_k > = < \tilde{\boldsymbol{\lambda}}_k, \Phi(k,j)\boldsymbol{\delta}\mathbf{x}_j > = < \Phi^T(k,j)\tilde{\boldsymbol{\lambda}}_k, \boldsymbol{\delta}\mathbf{x}_j > \tag{4.30}$$

$$= < \tilde{\boldsymbol{\lambda}}_j, \boldsymbol{\delta}\mathbf{x}_j >, \qquad \text{for all} \quad k \geq j \tag{4.31}$$

where $\tilde{\boldsymbol{\lambda}}_k$ and $\tilde{\boldsymbol{\lambda}}_j$ solve the unforced adjoint equations (4.26), and where $\boldsymbol{\delta}\mathbf{x}_j$ and $\boldsymbol{\delta}\mathbf{x}_k$ solve the tangent linear model (4.17).

4.2.2 Adjoint model development

Clearly, the derivation of the adjoint model is a major part in the setting up of the adjoint method. Here we outline a few different approaches to the derivation of the adjoint model which might be used in the wider context of optimization problems.

One approach is to work with a continuous, rather than a discrete version of the model. In this case, the calculus of variations is the appropriate theory for finding conditions for extrema of an optimization problem (we give some background on the calculus of variations, and an overview of this approach in the report [40]). Using the method of Lagrange multipliers (which in this case are functions) to deal with the model constraints, the adjoint model is given by the *Euler Lagrange equations*, and an expression analogous to equation (4.15) can be found for the gradient of the Lagrange functional with respect to the initial state. This leads us to a continuous analogue of Algorithm IS, involving the model equations, the adjoint equations and the gradient of the Lagrangian functional with respect to the initial state. In general, these will have to be discretized in some appropriate way so that the problem can be solved numerically. In application to data assimilation, this approach has been used in [6], for example.

A disadvantage of this approach, however, is that in general, the discretized adjoint equations will not in fact be the true adjoint equations of the discretized model equation [50]. Hence, the gradient calculated at each iteration will be inaccurate, and so it might not be possible to obtain a sufficiently accurate estimate of the optimal control vector.

The approach most generally taken in the development of adjoint models for data assimilation is that it is better to find the adjoint of the discrete model, so that theoretically at least, we can obtain an exact expression for the required gradient, [19]. We use this approach in the work described in this thesis, and work out the adjoint of the discrete models which we wish to use "by hand". For very large and complex models, however, this would be a much more difficult task.

A different approach to deriving the adjoint model is to work directly from the model computer code. Each assignment statement in the computer code can be treated as a constraint to be multiplied by a Lagrange multiplier. Differentiating each model statement with respect to the model variables gives the conditions on the Lagrange multipliers (or adjoint variables) which constitute the adjoint model. The paper by Chao and Chang [15] gives a different perspective on what it means to find the "adjoint" of computer code, and also gives a little more detail on the practical procedure of developing the adjoint code.

There is much research underway to produce computer software to automate the process of finding the adjoint of computer code. Developments in this field of *computational differentiation* [37], [11] are of particular interest for developing adjoint models in meteorology and oceanography, which is a tedious and error prone task.

4.3 The correction term technique

The method we refer to as the "correction term technique" was suggested by Derber [26]. In this approach, a constant correction term is used instead of, or as well as, the model initial state as the control vector in 4D varaitional assimilation. In Derber's paper, the technique is called "variational continuous assimilation", and the correction term was seen as a correction to the time derivatives of the model. In this approach, the correction made by the assimilation is evenly distributed over the entire assimilation interval, rather than concentrated at the initial time. This gives a solution of the data assimilation problem which is continuous from one assimilation interval to the next.

The other advantage of the correction term technique is that it can account for *schematic* model errors, and it is suggested in [26] that by application to many cases, this method could yield an estimate of the model's systematic error in each timestep. Further, the correction term found for an assimilation interval could be used in a subsequent forecast to counteract model error here too.

In the correction term technique, the model error is approximated by

$$\boldsymbol{\varepsilon}_k = s_k \mathbf{e}, \qquad k = 0, .., N-1 \tag{4.32}$$

where the $s_k \in \mathbb{R}$ are predetermined constants, and $\mathbf{e} \in \mathbb{R}^n$ is a constant *correction term* to be determined. Hence we work with the model

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + s_k \mathbf{e}_k, \qquad k = 0, .., N-1. \tag{4.33}$$

As before, we suppose we have observations

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1, \tag{4.34}$$

as defined in (2.6). The correction term is used as a control vector instead of, or as well as, the initial state.

Including the correction term $\mathbf{e}$ in the model equations, Problem $\mathcal{S}$ is modified to

Problem $\mathcal{CT}$

Minimize, with respect to $\mathbf{x}_0, .., \mathbf{x}_N, \mathbf{e}$

$$\mathcal{J} = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^T P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j)^T R_j^{-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j) \tag{4.35}$$

subject to (4.33).

We now summarize how the adjoint method for 4D variational assimilation using the initial state as a control vector can be modified to the correction term technique. In this case, $\mathbf{x}_0$ and $\mathbf{e}$ can be used as control vectors, since from them, the model states $\mathbf{x}_1, .., \mathbf{x}_N$ can be determined. Minimizing (4.35) with respect to the constraints (4.33) is equivalent to extremizing the Lagrangian function

$$\mathcal{L} = \mathcal{J} + \sum_{j=0}^{N-1} \boldsymbol{\lambda}_{j+1}^T(\mathbf{x}_{j+1} - \mathbf{f}_j(\mathbf{x}_j) - s_j \mathbf{e}). \tag{4.36}$$

Enforcing $\nabla_{\boldsymbol{\lambda}_k}\mathcal{L} = 0$ for $k = 1, .., N$ yields the model equations (4.33), and enforcing $\nabla_{\mathbf{x}_k}\mathcal{L} = 0$ for $k = 1, .., N$ yields the same adjoint equations as before

$$\boldsymbol{\lambda}_k = F^T_{\mathbf{x}_k}(\mathbf{x}_k)\boldsymbol{\lambda}_{k+1} - H^T_{\mathbf{x}_k}(\mathbf{x}_k)R_k^{-1}(\mathbf{h}_k(\mathbf{x}_k) - \mathbf{y}_k), \qquad k = 1, .., N-1, \quad (4.37)$$

$$\boldsymbol{\lambda}_N = 0. \quad (4.38)$$

As before, the gradient of $\mathcal{L}$ with respect to the initial state is

$$\nabla_{\mathbf{x}_0}\mathcal{L} = P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) - \boldsymbol{\lambda}_0, \quad (4.39)$$

and the gradient of $\mathcal{L}$ with respect to the correction term $\mathbf{e}$ is

$$\nabla_{\mathbf{e}}\mathcal{L} = -\sum_{j=1}^{N} s_{j-1}\boldsymbol{\lambda}_j. \quad (4.40)$$

Algorithm IS can easily be modified to use the correction term instead of, or as well as, the initial state as a control vector.

4.4 The weak constraint approach

We finally consider the weak constraint approach to 4D variational assimilation in which we allow for model error without the approximation that it is constant in time, and so we consider the model

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \boldsymbol{\varepsilon}_k, \qquad k = 0, .., N-1. \quad (4.41)$$

We again suppose that we have observations given by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k^t) + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1, \quad (4.42)$$

and a background estimate $\mathbf{x}_0^b$ satisfying

$$\mathbf{x}_0^t = \mathbf{x}_0^b + \boldsymbol{\beta}_0. \quad (4.43)$$

4.4.1 The general least squares problem

The classical *least squares* approach to estimating the true model state on the assimilation interval $[t_0, t_N]$ involves minimizing the errors $\boldsymbol{\beta}_0$, $\boldsymbol{\delta}_k$ and $\boldsymbol{\varepsilon}_k$ [8], and is as follows [44].

Problem $\mathcal{LS}$

Minimize, with respect to $\mathbf{x}_0, .., \mathbf{x}_N, \boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$

$$\begin{aligned} \mathcal{J} &= \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^T P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j)^T R_j^{-1}(\mathbf{h}_j(\mathbf{x}_j) - \mathbf{y}_j) \\ &+ \frac{1}{2}\sum_{j=0}^{N-1}\boldsymbol{\varepsilon}_j^T Q_j^{-1}\boldsymbol{\varepsilon}_j \end{aligned} \tag{4.44}$$

subject to (4.41),

where the symmetric, positive definite weighting matrices $P_0^{-1} \in \mathbb{R}^{n\times n}$, $R_k^{-1} \in \mathbb{R}^{p_k\times n}$ and $Q_k^{-1} \in \mathbb{R}^{n\times n}$ are based on our knowledge of the sizes of the errors $\boldsymbol{\beta}_0$, $\boldsymbol{\delta}_k$ and $\boldsymbol{\varepsilon}_k$ respectively. Problem $\mathcal{LS}$ is equivalent to the weak constraint minimization problem formulated by Sasaki [75].

Statistically optimal solutions to Problem $\mathcal{LS}$

We assume now that the model errors $\boldsymbol{\varepsilon}_k$, the observational errors $\boldsymbol{\delta}_k$ and the background error $\boldsymbol{\beta}_0$ are unbiased Gaussian random vectors, and that the matrices Q_k, R_k and P_0 are their respective error covariance matrices. We assume that the errors $\boldsymbol{\varepsilon}_k$ and $\boldsymbol{\delta}_k$ are not serially correlated, are uncorrelated with each other and with $\boldsymbol{\beta}_0$, and uncorrelated with the true model state. These are the assumptions made on model error and observational error in the standard Kalman filter described in Chapter 3, Section 3.2. In this case, minimizing (4.44) with respect to $\mathbf{x}_0, .., \mathbf{x}_N, \boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$ subject to the constraints (4.41) is equivalent to finding the *maximum likelihood Bayesian estimate* of $\mathbf{x}_0, .., \mathbf{x}_N$ given the observations $\mathbf{y}_0, .., \mathbf{y}_{N-1}$ and with prior estimate $\mathbf{x}_0^b$, which is given by the mode of the joint conditional pdf of $\mathbf{x}_0, .., \mathbf{x}_N$, [44].

If the model evolution described in (4.41) is linear, and the observations (4.42) are linearly related to the model state, then the conditional pdf of $\mathbf{x}_0, .., \mathbf{x}_N$ is Gaussian, and so unimodal. Further, in this case, the mode coincides with the mean, and hence the maximum likelihood Bayesian estimate is the same as the minimum variance estimate [44], [55]. In this case, a solution of Problem $\mathcal{LS}$ is a statistically optimal or "most likely" solution in the sense of both maximum likelihood and minimum variance.

In the nonlinear case, however, a cost function $\mathcal{J}$ with multiple minima corresponds to a pdf which is multimodal. As Jazwinski notes [44], maximum likelihood estimation is of questionable value unless the pdf is unimodal and concentrated about the mean.

We note that Problem $\mathcal{S}$ of Section 4.1 is a special case of Problem $\mathcal{LS}$ with the model error terms $\boldsymbol{\varepsilon}_k$ set fixed at zero. If the errors $\boldsymbol{\delta}_k$ and $\boldsymbol{\beta}_0$ are as specified above, then the strong constraint approach provides the statistically most likely solution if there is no model error. We discuss how the correction term technique can be interpreted statistically in Chapter 6.

4.4.2 Methods for solving Problem $\mathcal{LS}$

The purpose of this subsection is to overview the main approaches to solving a problem of the form of Problem $\mathcal{LS}$ which have been suggested for data assimilation in meteorology and oceanography. This highlights the well-known links between these methods, discussed for example in [55], [82] and [61], which we wish to exploit for a better understanding of how to deal with model error in 4D variational assimilation.

The Kalman filter

We outlined the standard Kalman filter for a linear model with observations linearly related to the model state, and under given statistical assumptions, in Chapter 3, Section 3.2. In that section, we stated that the Kalman filter solution $\mathbf{x}_k^a$ at time t_k is the most likely estimate of $\mathbf{x}_k^t$ given the observations $\mathbf{y}_0, .., \mathbf{y}_k$ and background estimate $\mathbf{x}_0^b$. In this linear case, the Kalman filter estimate is the same as the solution to Problem $\mathcal{LS}$ at time t_N, ie, at the end of the assimilation interval $[t_0, t_N]$, [44]. The *Kalman smoother* is a generalization of the Kalman filter which gives an estimate $\mathbf{x}_k^*$ at time t_k which is the most likely given the observations $\mathbf{y}_0, .., \mathbf{y}_N$, and so is equivalent to a solution of Problem $\mathcal{LS}$, [61] although this requires still greater cost.

One major advantage of the Kalman filter as a method of solving Problem $\mathcal{LS}$ in the linear case is that it produces at each timestep the error covariance matrix of the analysed state. Hence, at time t_N, we have not only the optimal estimate of the state, but also its error covariance matrix. This provides the background error

covariance matrix P_0 for the next assimliation interval, and also gives us a way of calculating the accuracy of a forecast initiated at time t_N.

In this advantage of the Kalman filter lies also (arguably) the biggest drawback for its application in operational assimilation in meteorology and oceanography, compared to other methods of finding the same statistically optimal solution. This drawback is the huge cost of propagating the error covariance matrices in time, which involves $O(n^2)$ operations [32]. Hence for operational meteorological models, where the dimension n of the state is $O(10^5) - O(10^7)$, the Kalman filter in unsimplified form is considered too expensive [32], [72]. Various simplifications of the Kalman filter have been proposed, however, [84]. We note that several centres planning to implement variational assimilation schemes operationally are also proposing the use of a simplified Kalman filter to solve the problem of specifying the covariance matrix P_0 for the beginning of a new assimilation interval.

The representer method

The representer method was suggested for oceanographic applications of data assimilation by Bennett et. al. [7] and for meteorological assimilation by Bennett et. al. [6] and by Amodei [1]. For a linear system, it provides a method for solving the *Euler-Lagrange equations* which constitute necessary conditions for a solution of a continuous analogue of Problem $\mathcal{LS}$. The method involves iterating on elements of the "space of observations". This is similar in principal to the PSAS algorithm outlined in Chapter 3, Section 3.1, which is for analysis at a single time only. In the nonlinear case, solutions to the (nonlinear) Euler-Lagrange equations are found by solving a sequence of linear Euler-Lagrange equations using the representer method.

Since the dimension of the observation vector is generally much smaller than the dimension of the state vector at any time, this method is potentially much more efficient than other approaches to solving Problem $\mathcal{LS}$. In oceanography, where the number of observations is $O(10^3)$, the potential advantage of this method is greater than in meteorology. In meteorology, where the number of observations at just one time might be $O(10^5)$, this method is still too expensive [5], but might become feasible in the future.

The adjoint method

We consider here the technique of reducing the control vector for solving Problem $\mathcal{LS}$. In this case, we have $N+1$ state vectors plus N model error vectors as variables, and N sets of model constraints (4.41). If we use $\mathbf{x}_0$ and $\boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$ as $N+1$ control vectors, we can uniquely determine the remaining variables, the N state vectors $\mathbf{x}_1, .., \mathbf{x}_N$ from the N constraints (4.41). The constrained minimization problem Problem $\mathcal{LS}$ is equivalent to the unconstrained optimization problem of extremizing the Lagrangian function

$$\mathcal{L} = \mathcal{J} + \sum_{j=0}^{N-1} \boldsymbol{\lambda}_{j+1}^T (\mathbf{x}_{j+1} - \mathbf{f}_j(\mathbf{x}_j) - \boldsymbol{\varepsilon}_j), \tag{4.45}$$

where $\boldsymbol{\lambda}_1, .., \boldsymbol{\lambda}_N \in \mathbb{R}^n$ are N vectors of Lagrange multipliers.

Using the method of reducing the control vectors described in Chapter 2, Section 2.3, the solution can be found by iterating on the control vectors $\mathbf{x}_0$ and $\boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$, as follows. From a guess of the control vectors, the model states $\mathbf{x}_1, .., \mathbf{x}_N$ are calculated using the constraints (4.41). As before, this ensures that $\nabla_{\boldsymbol{\lambda}_k} \mathcal{L} = 0$ for $k = 1, .., N$, and again, enforcing $\nabla_{\mathbf{x}_k} \mathcal{L} = 0$ for $k = 1, .., N$, yields the same adjoint equations

$$\boldsymbol{\lambda}_k = F_k^T(\mathbf{x}_k)\boldsymbol{\lambda}_{k+1} - H_k^T(\mathbf{x}_k) R_k^{-1} (\mathbf{h}_k(\mathbf{x}_k) - \mathbf{y}_k), \qquad k = 1, .., N-1, \tag{4.46}$$

$$\boldsymbol{\lambda}_N = 0. \tag{4.47}$$

As before, the gradient with respect to the initial state is given by

$$\nabla_{\mathbf{x}_0} \mathcal{L} = P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) - \boldsymbol{\lambda}_0, \tag{4.48}$$

where the additional adjoint vector $\boldsymbol{\lambda}_0 \in \mathbb{R}^n$ is defined via the relation (4.46) with $k = 0$. The gradient with respect to the model error vector $\boldsymbol{\varepsilon}_k$ for $k = 0, .., N-1$ is given by

$$\nabla_{\boldsymbol{\varepsilon}_k} \mathcal{L} = Q_k^{-1} \boldsymbol{\varepsilon}_k - \boldsymbol{\lambda}_{k+1}, \qquad k = 0, .., N-1. \tag{4.49}$$

These gradients can be used in a descent algorithm, such as one outlined in Chapter 2, Section 2.4, to improve our guess of the control vectors. We summarize this procedure in the following algorithm.

Algorithm ISME

1. From a guess of the control vectors $\mathbf{x}_0$ and $\boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$,
   calculate the model states $\mathbf{x}_1, .., \mathbf{x}_N$ using the model equations (4.41).

2. From the end condition (4.47), calculate the adjoint vectors $\boldsymbol{\lambda}_{N-1}, .., \boldsymbol{\lambda}_0$ using the model states calculated in Step 1.

3. From $\boldsymbol{\lambda}_{N-1}, .., \boldsymbol{\lambda}_0$, calculate $\nabla_{\boldsymbol{\varepsilon}_{N-1}}\mathcal{L}, .., \nabla_{\boldsymbol{\varepsilon}_0}\mathcal{L}$ and $\nabla_{\mathbf{x}_0}\mathcal{L}$ using (4.49) and (4.48).

4. Use these gradients in a gradient algorithm to obtain a better guess of the control vectors $\mathbf{x}_0$,and $\boldsymbol{\varepsilon}_0, .., \boldsymbol{\varepsilon}_{N-1}$, and repeat until convergence criteria are satisfied.

In practice, this algorithm is expensive for operational data assimilation, since it involves iterating on N control vectors of dimension n. A second problem is that the conditioning of the problem minimizing the cost function simultaneously with all these control vectors could be very poor, [5], [83].

We mention here, however, an attempt to solve a similar problem by Thacker and Long [83] in the context of a simple oceanographic model. Rather than trying to recover the model error vectors $\boldsymbol{\varepsilon}_k$, they attempt to recover unknown model forcing terms (wind stresses) from the data, although they mention that model error is accounted for via uncertainty in the forcing. They look at the question of data sufficiency, and find that if forcing is to be recovered with the initial state, a huge amount of extra data is needed, much more than they could expect to be available in practice. They also mention that the problem of recovering model forcing with the initial state is ill-conditioned and requires many iterations of the descent algorithm.

Chapter 5

The correction term technique

We described the "correction term technique" for 4D variational assimilation introduced by Derber [26] in Chapter 4 Section 3. Here we take a further look at both theoretical and practical aspects of using a correction term as a control vector, instead of or as well as using the initial state as a control vector.

In Section 5.2, we look for conditions for uniqueness of the solution to the 4D variational assimilation problem using the initial state, the correction term and both together as control vectors. Using the initial state as the control vector, uniqueness depends on the condition of complete N-step observability of the system. We show, however, that in general conditions for a unique solution using the correction term as a control vector are different, and so it might be possible to determine uniquely the initial state from the data and not the correction term, or vice versa. In each case adding a background estimate of the control vector to the cost function guarantees uniqueness in the case of data insufficiency. This point has been considered in data assimilation using the initial state as the control vector [8], but not in published work using the correction term technique. We look at uniqueness of the solution using both control vectors together by using the technique of state augmentation, and by relating conditions for observability of the augmented system with conditions for observability of the original system.

In a practical context, we compare the performance of the different control vectors using a simple linear model. We compare the ability of each control vector to compensate for errors in the initial state and for model error which is constant in

time. We also examine the impact of the number of observations on the results, and the use of a background estimate of the correction term. The experiments are described in Section 5.3, and the results are presented and discussed in Section 5.4. In Section 5.5 we summarize the theoretical and practical results of the chapter.

In Chapter 6 we extend the ideas given here on accounting for model error in 4D variational assimilation, and in Chapter 7 we discuss further how each of these control vectors can account for different types of model error in the context of a less simple model.

5.1 Background

In this section, we summarize use of the correction term technique in meteorology, and highlight research areas.

5.1.1 Use of the technique in the literature

The correction term technique, which we described in Chapter 4, Section 4.3, was suggested by Derber [26]. The experiments described in this paper showed better results were obtained using the correction term than were obtained using the initial state as the control vector. It was acknowledged that the comparative success of the correction term technique might in this case have been partly due to the fact that the model used was known to be inaccurate. It was also pointed out, however, that since 4D variational assimilation might be performed using simplified models, the correction term approach has potential. It was mentioned briefly that an attempt at using both control vectors simultaneously had not been successful. In this paper, the cost function to be minimized penalizes distance to the observations only.

The correction term technique was later applied by M. Zupanski, [94], who interpreted the correction term as representing *model bias*. The experiments carried out were on a full regional forecast model, but with approximate adjoint equations (and hence inaccurate gradient calculations). The cost function consisted of two terms, one to penalize distance from observed data (which came from OI analyses), and the other to penalize spurious gravity waves. The results showed that using the

correction term as the control vector worked better than using the initial state as the control vector.

M. Zupanski [94] extended Derber's work by trying to use both control vectors. Using both simultaneously did not work as well as hoped, which was believed to be because of problems with preconditioning. Better results, as characterised by greater cost function reduction during the assimilation and greater reductions of rms errors in the ensuing forecast, were claimed by using first the initial state and then the correction term as a control vector. It was acknowledged, however, that minimizing the cost function first with respect to one control vector and then with respect to the other constitutes a different optimization problem to minimizing the cost function with respect to both control vectors simultaneously. As in Derber's paper [26], it was found that using the correction term determined by the assimilation in a subsequent forecast improved the forecast.

The correction term technique was also used by D. Zupanski and Mesinger [93] in a paper primarily on the problems of assimilating precipitation data. Here the correction term and initial state were used simultaneously as control vectors. This paper does not give details on experiments with different control vectors, but mentions that using both control vectors reduced the cost function 20 percent more than using just the initial state as a control vector. Here, the largest components of the correction term recovered were nearest to the model boundaries.

The correction term technique was also investigated by Wergen [87] in a more idealized context, using a linearized one-dimensional shallow water model. The paper considers more generally the impact of model error on variational assimilation, and gives a very interesting discussion on allowing for model error in variational assimilation. Using the initial state and the correction term simultaneously, it was found that the correction term could very successfully recover omitted *constant* forcing terms. In this case, using the recovered forcing terms in the ensuing forecast was also successful. In this context, Wergen refers to the correction term technique as *variational tuning*, and points out that this approach provides a way to tune simultaneously several model parameters and so obtain the proper interactions between the various parameters.

In the experiments of Wergen's paper, an extra term, constraining the correction to the mass field to be small, was added to the cost function. The aim was to make the approach more like Sasaki's weak constraint method [75], but the mass field only and not the other two fields was constrained in this way, "for simplicity". The other papers mentioned above included no constraint on the correction term in the cost function. In rather testing experiments in which the model error consisted of a 50 percent phase error in the Rossby modes, using both control vectors together gave better results than using the initial state only during the assimilation period of 24 hours. The results of the ensuing forecast were also improved for the first 12 hours, but after this were worse than if just the initial state was used as the control vector.

From these results Wergen points out the danger that allowing for model error in variational assimilation allows freedom which could yield a solution which is close to the data but physically unrealistic, and that use of the correction terms in a subsequent forecast will be detrimental if they do not compensate for the real model error. He concludes that the problem of how to incorporate statistical information on model error into variational assimilation, in a way consistent with the Kalman filter, is a very important issue.

5.1.2 Research issues

We now discuss some issues on using the correction term as a control vector that are worth further investigation. Firstly, no background estimate for the correction term is used in the work described above. It is known however, that when the initial state is used as a control vector, including a background estimate of this is essential to guarantee a unique solution in certain cases [8]. We might ask, under what conditions is it necessary to use a background estimate of the correction term? Further, are these conditions the same as when a background estimate for the initial state is necessary?

Another question is, if a background estimate for the correction term is included in the cost function, how should this be weighted against the other terms in the cost function? In particular, how can this weighting be chosen so that the solution will be statistically optimal?

Moving on from these theoretical issues, Wergen's question [87] of whether the constant correction term can "compensate for the real model error" is worth further investigation. This is important, since using the correction term technique means altering the model equations in some way. Therefore it would be worth checking whether the correction term found in the assimilation seems to be a credible representation of model error. This could mean checking the size of the correction term, checking which model variables have been corrected, and which locations the corrections refer to.

We might expect the constant correction term to compensate well for constant model errors, but how well does it compensate for model errors which are not constant, especially model error which is known to depend on the model state? Perhaps a useful question to address is, on what timescale can the correction term correct for model error? It might also be profitable to investigate how we can modify the correction term technique to better deal with model error which is not constant in time.

Finally, in the research described above, using both the initial state and the correction term together as control vectors was not always successful. It is important to find preconditioning to improve the efficiency of the method using both control vectors. It is also worth investigating the importance of a background estimate for each of the control vectors in this case, too.

In the remainder of this chapter, and in Chapters 6 and 7, we address some of these issues both theoretically and also practically using simple models.

5.2 Uniqueness and observability

In this chapter, we consider theory only for the linear case. At first this may seem restrictive, since in an operational context, models are generally nonlinear and observational data is often nonlinearly related to the model state. However, the 4D variational assimilation problem is generally being planned for implementation using the incremental approach, in which the full problem is reduced to the minimization of a quadratic function with linear constraints (or a series of such minimizations).

This approach is justified because of the validity of the tangent linear model over the assimilation length scales and is necessary because of limitations of computational resources (Chapter 4, Section 4.1). Further, and importantly, the incremental approach gives us a minimization problem with a unique solution under certain conditions which we specify here.

We consider the general linear model (2.8). For convenience, we suppose there is no forcing of the form $B_k\mathbf{u}_k$; this does not alter the results of the theory but avoids unnecessary complication. Hence, we suppose the true model state satisfies

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + \boldsymbol{\varepsilon}_k. \tag{5.1}$$

We approximate model error by

$$\boldsymbol{\varepsilon}_k = B_k\mathbf{e} \tag{5.2}$$

where $\mathbf{e} \in \mathbb{R}^m$ is the *correction term*, and the matrices $B_k \in \mathbb{R}^{n\times m}$ are prescribed, with $\mathrm{Rank}(B_k) = m$. Hence the model we use for assimilation is

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k\mathbf{e}. \tag{5.3}$$

In the method proposed by Derber [26], the correction term $\mathbf{e}$ has the same dimension as the model state, and the matrices B_k are replaced by prescribed scalars s_k. Introducing the matrices B_k however allows us to use a correction term with dimension m less than the state dimension n. This could lead to increased efficiency if we know in advance that model error is concentrated in certain locations.

As before, we have observations available over N timesteps related to the true model state by

$$\mathbf{y}_k = C_k\mathbf{x}_k^t + \boldsymbol{\delta}_k, \qquad k = 0, \ldots N-1, \tag{5.4}$$

as defined in (2.9).

We now formulate the general data assimilation problem we wish to address in this chapter, that of estimating the model states and constant input over the assimilation interval, using the observational data (5.4).

Problem $\mathcal{A}_{ISCT}$

Minimize with respect to $\mathbf{x}_0, .., \mathbf{x}_N; \mathbf{e}$,

$$\mathcal{J} = \frac{1}{2} \sum_{j=0}^{N-1} (C_j \mathbf{x}_j - \mathbf{y}_j)^T R_j^{-1} (C_j \mathbf{x}_j - \mathbf{y}_j) \tag{5.5}$$

subject to (5.3).

In (5.5) the matrices $R_j^{-1} \in \mathbb{R}^{p_j \times p_j}$ are assumed to be symmetric positive definite, and to represent the relative accuracies of the observational data. Ideally, they should be the inverse observational error covariance matrices. We consider modifications of Problem $\mathcal{A}_{ISCT}$ involving background terms later.

Since $\text{Rank}(A_k) = n$ and $\text{Rank}(B_k) = m$ for all k by assumption, specification of $\mathbf{x}_0$ and $\mathbf{e}$ uniquely determines the model state at all subsequent times. Hence $\mathbf{x}_0$ and $\mathbf{e}$ can be used as control vectors, and we view Problem $\mathcal{A}_{ISCT}$ as that of finding an optimal initial state (IS) and correction term (CT).

If we consider the correction term to be fixed (for example, if we assume that there is no model error, or that model error is represented by a known bias), the initial state is the control variable and we have the familiar strong constraint 4D variational assimilation method outlined in Chapter 4, Section 4.1. Here we will refer to this as Problem $\mathcal{A}_{IS}$.

Problem $\mathcal{A}_{IS}$

Minimize $\mathcal{J}$ *defined in (5.5) with respect to* $\mathbf{x}_0, ...\mathbf{x}_N$, *subject to (5.3), with* $\mathbf{e}$ *fixed.*

The problem addressed by correction term technique, using the correction term only as the control vector, we refer to here as Problem $\mathcal{A}_{CT}$. In this case we assume the initial state is known.

Problem $\mathcal{A}_{CT}$

Minimize $\mathcal{J}$ *defined in (5.5) with respect to* $\mathbf{x}_1, ...\mathbf{x}_N; \mathbf{e}$, *subject to (5.3), with* $\mathbf{x}_0$ *fixed.*

One of our objectives in this chapter is to give conditions under which Problems $\mathcal{A}_{ISCT}$, $\mathcal{A}_{IS}$ and $\mathcal{A}_{CT}$ have a unique solution.

Our approach here, rather than using Lagrange multipliers to reduce the problem to an unconstrained problem is to substitute the model constraints (5.3) directly into the cost function. We do this for theoretical purposes in this chapter, but use the Lagrange multiplier technique for our practical examples. We express the state $\mathbf{x}_k$ at time t_k in terms of the control vectors $\mathbf{x}_0$ and $\mathbf{e}$, which is possible using the state transition matrix.

From Chapter 2 equation (2.16) we have

$$\mathbf{x}_k = \Phi(k,0)\mathbf{x}_0 + \sum_{j=0}^{k-1} \Phi(k,j+1)B_j\mathbf{e}, \qquad k = 1,..N, \tag{5.6}$$

where the state transition matrix is given by

$$\Phi(k,j) = \prod_{i=j}^{k-1} A_i, \qquad k > j, \tag{5.7}$$

$$\Phi(j,j) = I. \tag{5.8}$$

For convenience, we write (5.6) as

$$\mathbf{x}_k = \Phi_k\mathbf{x}_0 + \tilde{B}_k\mathbf{e}, \qquad k = 0,..N, \tag{5.9}$$

where

$$\tilde{B}_k = \sum_{j=0}^{k-1} \Phi(k,j+1)B_j, \qquad k = 1,..N, \tag{5.10}$$

$$\tilde{B}_0 = 0, \tag{5.11}$$

and

$$\Phi_k = \Phi(k,0), \qquad k = 0,..N. \tag{5.12}$$

Incorporating the model constraints, we find that the cost function $\mathcal{J}$ can be written in terms of $\mathbf{x}_0$ and $\mathbf{e}$ as

$$\tilde{\mathcal{J}} = \frac{1}{2}\sum_{j=0}^{N-1} (C_j(\Phi_j\mathbf{x}_0 + \tilde{B}_j\mathbf{e}) - \mathbf{y}_j)^T R_j^{-1}(C_j(\Phi_j\mathbf{x}_0 + \tilde{B}_j\mathbf{e}) - \mathbf{y}_j), \tag{5.13}$$

which after manipulation gives

$$\begin{aligned}\tilde{\mathcal{J}} \;=\; & \frac{1}{2}\sum_{j=0}^{N-1} \{\mathbf{x}_0^T\Phi_j^T C_j^T R_j^{-1} C_j\Phi_j\mathbf{x}_0 + \mathbf{e}^T\tilde{B}_j^T C_j^T R_j^{-1} C_j\tilde{B}_j\mathbf{e} \\ & +2(C_j\Phi_j\mathbf{x}_0 - \mathbf{y}_j)^T R_j^{-1}(C_j\tilde{B}_j\mathbf{e} - \mathbf{y}_j) - \mathbf{y}_j^T R_j^{-1}\mathbf{y}_j\}.\end{aligned} \tag{5.14}$$

This can be verified by noting that for any vectors $\mathbf{a},\mathbf{b},\mathbf{c} \in \mathbb{R}^n$, the following identity holds.

$$(\mathbf{a}+\mathbf{b}-\mathbf{c})^T(\mathbf{a}+\mathbf{b}-\mathbf{c}) = \mathbf{a}^T\mathbf{a} + \mathbf{b}^T\mathbf{b} + 2(\mathbf{a}-\mathbf{c})^T(\mathbf{b}-\mathbf{c}) - \mathbf{c}^T\mathbf{c}. \tag{5.15}$$

5.2.1 Using the initial state as the control vector

In this subsection we consider Problem $\mathcal{A}_{IS}$, which consists of constrained minimization of (5.5) subject to (5.3), with $\mathbf{e}$ fixed, so throughout this subsection $\mathbf{e}$ is assumed to be given.

Our aim is to find conditions for a unique minimum $\mathbf{x}_0$ of Problem $\mathcal{A}_{IS}$. In terms of the initial state $\mathbf{x}_0$, (5.14) can be written

$$\tilde{\mathcal{J}} = \frac{1}{2}\mathbf{x}_0^T \tilde{A}_{IS}\mathbf{x}_0 + \tilde{\mathbf{b}}_{IS}^T\mathbf{x}_0 + \tilde{c}_{IS}, \tag{5.16}$$

where $\tilde{\mathbf{b}}_{IS} \in \mathbb{R}^n$, $\tilde{c}_{IS} \in \mathbb{R}$, and the Hessian matrix $\mathcal{A}_{IS} \in \mathbb{R}^{n\times n}$ is given by

$$\tilde{A}_{IS} = \sum_{j=0}^{N-1} \Phi_j^T C_j^T R_j^{-1} C_j \Phi_j. \tag{5.17}$$

From the theory of Chapter 2, Section 2.3, a necessary condition for $\mathbf{x}_0$ to be a minimum is that $\nabla_{\mathbf{x}_0}\tilde{\mathcal{J}}$ vanishes, ie,

$$\tilde{A}_{IS}\mathbf{x}_0 + \tilde{\mathbf{b}}_{IS} = 0. \tag{5.18}$$

Any $\mathbf{x}_0$ satisfying (5.18) is a minimum since $\tilde{A}_{IS}$ is positive semi-definite, and it is unique if and only if $\tilde{A}_{IS}$ is positive definite or equivalently if and only if $\text{Rank}(\tilde{A}_{IS}) = n$. We now link this condition of uniqueness to the observability of the system.

The observations (5.4) may be related to $\mathbf{x}_0$ and $\mathbf{e}$ using (5.9) as follows.

$$\mathbf{y}_k = C_k(\Phi_k\mathbf{x}_0 + \tilde{B}_k\mathbf{e}) + \boldsymbol{\delta}_k, \qquad k = 0,..,N-1, \tag{5.19}$$

which may be written as

$$\mathcal{O}_N\mathbf{x}_0 = Y_N - \mathcal{T}_N\mathbf{e} + D_N, \tag{5.20}$$

where

$$\mathcal{O}_N = \begin{pmatrix} C_0\Phi_0 \\ C_1\Phi_1 \\ \vdots \\ C_{N-1}\Phi_{N-1} \end{pmatrix}, \qquad \mathcal{T}_N = \begin{pmatrix} 0 \\ C_1\tilde{B}_1 \\ \vdots \\ C_{N-1}\tilde{B}_{N-1} \end{pmatrix}, \tag{5.21}$$

and

$$Y_N = \begin{pmatrix} \mathbf{y}_0 \\ \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_{N-1} \end{pmatrix}, \qquad D_N = \begin{pmatrix} \boldsymbol{\delta}_0 \\ \boldsymbol{\delta}_1 \\ \vdots \\ \boldsymbol{\delta}_{N-1} \end{pmatrix}. \tag{5.22}$$

We note that $\mathcal{O}_N = \mathcal{O}_N^0$, the N-step observability matrix at time t_0 defined in equation (2.20).

If there is no observational error, then the right hand side of (5.20) is known. In Chapter 2, Section 2.2, we defined *complete N-step observability at time t_0* as the ability to determine uniquely the state $\mathbf{x}_0$ from the observations $\mathbf{y}_k$ and specified inputs $B_k\mathbf{e}$, $k = 0, .., N-1$. It was proved in Theorem 2.2 that the system is completely N-step observable at time t_0 if and only if $\text{Rank}(\mathcal{O}_N) = n$, and therefore N-step observability is a necessary and sufficient condition to determine a unique initial state $\mathbf{x}_0$ in this case. We note, however, that only if the model (5.3) is a perfect representation of the evolution of the true model state $\mathbf{x}_k^t$, will the solution $\mathbf{x}_0$ of (5.20) represent the true initial state $\mathbf{x}_0^t$.

In practice, the observational error is not negligible. It is therefore not possible in general to estimate $\mathbf{x}_0$ exactly from the data, since the observational errors are unknown. Hence we attempt to find a least-squares estimate for $\mathbf{x}_0$, which can be done by solving Problem $\mathcal{A}_{IS}$. If the observational errors $\boldsymbol{\delta}_k$ are unbiased, Gaussian and uncorrelated in time with covariance matrices $\text{Cov}\{\boldsymbol{\delta}_k, \boldsymbol{\delta}_k\} = R_k$, and if the model (5.3) is a perfect representation of the evolution of the true model state, then this least squares estimate of $\mathbf{x}_0$ is the *most likely* estimate of $\mathbf{x}_0^t$. If these assumptions on the observational error statistics and on the model accuracy do not hold to a good approximation, then our estimate $\mathbf{x}_0$ will not be a good approximation to the most likely estimate.

The question of whether Problem $\mathcal{A}_{IS}$ has a unique solution depends on complete N-step observability, as we now show.

Definition 5.1 *We say that the linear time varying system (5.3),(5.4) containing observational error is completely N-step observable at time t_j if the corresponding system with no observational error is completely N-step observable at time t_j.*

Theorem 5.1 *Problem $\mathcal{A}_{IS}$ has a unique solution if and only if the system (5.3),(5.4) is completely N-step observable at time t_0.*

Proof

From the previous discussion, it suffices to show that Rank $(\tilde{A}_{IS}) = n$ if and only if Rank $(\mathcal{O}_N) = n$.

Since the matrices R_j^{-1} are positive definite, they can be written uniquely as follows

$$R_j^{-1} = U_j^T U_j, \tag{5.23}$$

where $U_j \in \mathbb{R}^{p_j \times p_j}$ are positive definite matrices.

Defining $\tilde{U}$ to be the positive definite block diagonal matrix with block elements $U_0, .., U_{N-1}$, ie

$$\tilde{U} = \begin{pmatrix} U_0 & & & \\ & U_1 & & \\ & & \ddots & \\ & & & U_{N-1} \end{pmatrix}, \tag{5.24}$$

we have

$$\tilde{A}_{IS} = \mathcal{O}_N^T \tilde{U}^T \tilde{U} \mathcal{O}_N. \tag{5.25}$$

Suppose that Rank$(\tilde{A}_{IS}) < n$. Then there exists a non-zero vector $\mathbf{v} \in \mathbb{R}^n$ such that

$$\mathbf{v}^T \tilde{A}_{IS} \mathbf{v} = 0, \tag{5.26}$$

$$\Rightarrow \tilde{U} \mathcal{O}_N \mathbf{v} = 0 \tag{5.27}$$

$$\Rightarrow \text{Rank}(\mathcal{O}_N) < n, \tag{5.28}$$

since $\tilde{U}$ is positive definite.

Similarly, suppose Rank$(\mathcal{O}_N) < n$. Then there exists a non-zero vector $\mathbf{u} \in \mathbb{R}^n$ such that

$$\mathcal{O}_N \mathbf{u} = 0, \tag{5.29}$$

$$\Rightarrow \mathbf{u}^T \tilde{A}_{IS} \mathbf{u} = 0, \tag{5.30}$$

$$\Rightarrow \text{Rank}(\tilde{A}_{IS}) < n. \tag{5.31}$$

Hence, Rank$(\tilde{A}_{IS}) = n$ if and only if Rank$(\mathcal{O}_N) = n$. □

Theorem 5.1 is a recasting for our data assimilation problem of a known result in filtering theory that the time-varying system (5.3),(5.4) is completely observable if and only if $\tilde{A}_{IS}$ is positive definite for some N, [44].

We now specialize to the time-invariant case, which is given by

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + B\mathbf{e}, \tag{5.32}$$

$$\mathbf{y}_k = C\mathbf{x}_k + \boldsymbol{\delta}_k. \tag{5.33}$$

The following Corollary links the concept of *complete observability* of the time invariant system (5.32),(5.33) to uniqueness of Problem $\mathcal{A}_{IS}$. We note however, that Theorem 5.1 also applies to the time invariant case.

Definition 5.2 *We say that the linear time varying system (5.32),(5.33) is completely observable if the corresponding system with no observational error is completely observable.*

Corollary 1 *For the time invariant system (5.32),(5.33), if $N \geq n$ then Problem $\mathcal{A}_{IS}$ has a unique solution if and only if the system (5.32),(5.33) is completely observable.*

Proof

This result follows from Theorem 5.1, since if $N \geq n$, complete N-step observability at time t_0 of a time invariant system is equivalent to complete observability of that system, by Theorem 2.3, Part b. □

The paper by Zou, Navon and Le Dimet [91] included a proof for the continuous, time invariant case that complete observability is a sufficient condition for a unique solution of the problem, and stated that similar results may be obtained for a discrete model. In the discrete case, however, the number N of timesteps in the assimilation interval is important to whether the problem has a unique solution. Corollary 5.1

does not tell us whether complete observability is sufficient for uniqueness if $N < n$, but Theorem 5.1 does. Since in most applications of data assimilation in meteorology and oceanography, the number of timesteps over which observations are available is far less than the dimension of the system (ie, $N << n$), this is an important point. Hence, using the concept of complete N-step observability is important for our application not only because it allows generalization to a time-varying system, but also because it can be used to give a condition for uniqueness of the data assimilation problem that depends on the length of the assimilation interval, and the particular time t_0 at which the assimilation is started.

We now show that if the system is not completely N-step observable at time t_0, we can ensure a unique solution by adding a background term to the cost function. It has often been stated in the data assimilation literature that a background term can make up for data insufficiency [6], [73], [55]. Bennett and Miller [8] show for a linear model, expressed in terms of Fourier coefficients, that a background estimate of the initial state is sufficient for uniqueness. They also argue that unless there is sufficient independent data, such a background term is in fact necessary for uniqueness.

We consider the following minimization problem.

Problem $\mathcal{B}_{IS}$

Minimize with respect to $\mathbf{x}_0, .., \mathbf{x}_N$

$$\mathcal{J} = \frac{1}{2}(\mathbf{x}_0 - \mathbf{x}_0^b)^T P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(C_j\mathbf{x}_j - \mathbf{y}_j)^T R_j^{-1}(C_j\mathbf{x}_j - \mathbf{y}_j) \qquad (5.34)$$

subject to (5.3) with $\mathbf{e}$ *fixed.*

In (5.34), $\mathbf{x}_0^b \in \mathbb{R}^n$ is a background guess for $\mathbf{x}_0$, and $P_0^{-1} \in \mathbb{R}^{n\times n}$ is a symmetric positive definite weighting matrix, ideally approximating the inverse covariance matrix of the errors $(\mathbf{x}_0 - \mathbf{x}_0^b)$. Equation (5.34) can be written

$$\tilde{\mathcal{J}} = \frac{1}{2}\mathbf{x}_0^T(P_0^{-1} + \tilde{A}_{IS})\mathbf{x}_0 + (\tilde{\mathbf{b}}_{IS} - \mathbf{x}_0^b P_0^{-1})^T\mathbf{x}_0 + \tilde{c}_{IS} + \frac{1}{2}(\mathbf{x}_0^b)^T P_0^{-1}\mathbf{x}_0^b, \qquad (5.35)$$

so in this case the Hessian of $\tilde{\mathcal{J}}$ with respect to $\mathbf{x}_0$ is $(P_0^{-1} + \tilde{A}_{IS})$. We have

Theorem 5.2 *Problem $\mathcal{B}_{IS}$ has a unique solution.*

Proof

The Hessian matrix $(P_0^{-1} + \tilde{A}_{IS})$ is the sum of a positive definite and a positive semi-definite matrix, and hence is positive definite. □

5.2.2 Using the correction term as the control vector

Here we consider the case where $\mathbf{x}_0$ is fixed and the correction term $\mathbf{e}$ is the control vector, and seek conditions for a unique solution of Problem $\mathcal{A}_{CT}$.

In this case, we wish to express the cost function $\mathcal{J}$ in terms of the correction term. From (5.14) we have

$$\tilde{\mathcal{J}} = \frac{1}{2}\mathbf{e}^T \tilde{A}_{CT}\mathbf{e} + \tilde{\mathbf{b}}_{CT}^T\mathbf{e} + \tilde{c}_{CT} \tag{5.36}$$

where $\tilde{\mathbf{b}}_{CT} \in \mathbb{R}^m$, $\tilde{c}_{CT} \in \mathbb{R}$, and $\tilde{A}_{CT} \in \mathbb{R}^{m\times m}$ is the Hessian of $\tilde{\mathcal{J}}$ with respect to $\mathbf{e}$, given by

$$\tilde{A}_{CT} = \sum_{j=0}^{N-1} \tilde{B}_j^T C_j^T R_j^{-1} C_j \tilde{B}_j. \tag{5.37}$$

Hence, Problem $\mathcal{A}_{CT}$ has a unique solution if and only if $\tilde{A}_{CT}$ is positive definite.

In Subsection 5.2.1, we related the observational data to the initial state and correction term. In the case that observational error can be neglected, we have from equation (5.20),

$$\mathcal{T}_N\mathbf{e} = Y_N - \mathcal{O}_N\mathbf{x}_0. \tag{5.38}$$

By analogous arguments to those given in the proof of Theorem 2.2, if $\mathbf{x}_0$ is known, then $\mathbf{e}$ satisfying (5.3) can be uniquely determined from the observations if and only if $\text{Rank}(\mathcal{T}_N) = m$. However, unless (5.3) is a perfect representation of the true model evolution, then the solution obtained using this value of $\mathbf{e}$ does not represent the true model state.

We note that (5.3) now has the same form as the general linear model (2.7), in which the time-varying input $\mathbf{u}_k$ has been replaced by the constant input $\mathbf{e}$. In the context of control theory, the problem of determining unknown or required model inputs from the outputs is referred to as *system inversion*, and this has been studied since the late 1960's [71]. Some theory for the time invariant continuous case is

given in [70], [71], and [64]. Our problem is rather different, however, since we only look for a constant input, and because we consider the discrete, time-varying case.

Since observational errors are not negligible in reality, we seek a least squares estimate of the correction term $\mathbf{e}$ from the observational data by solving Problem $\mathcal{A}_{CT}$. We now give conditions under which this problem has a unique solution.

Theorem 5.3 *Problem $\mathcal{A}_{CT}$ has a unique solution if and only if Rank$(\mathcal{T}_N) = m$.*

Proof

With the matrix $\tilde{U}$ defined as in (5.24), we have

$$\tilde{A}_{CT} = \mathcal{T}_N^T \tilde{U}^T \tilde{U} \mathcal{T}_N. \tag{5.39}$$

Since $\tilde{U}$ is positive definite, we have, by the same argument as in Theorem 5.1, that

$$\text{Rank}(\tilde{A}_{CT}) = m \quad \text{if and only if} \quad \text{Rank}(\mathcal{T}_N) = m. \tag{5.40}$$

□

It is interesting to note, however, that complete N-step observability is neither a necessary nor a sufficient condition for a unique solution of Problem $\mathcal{A}_{CT}$. Hence, given the same set of observations, it is possible that Problem $\mathcal{A}_{IS}$ has a unique solution but Problem $\mathcal{A}_{CT}$ does not, and vice versa. We show this by means of simple counter-examples, in the case where $m = n$ and the matrices B_k are equal to the identity matrix. We note that the result does not rely on the fact that the observations $\mathbf{y}_0$ contain information about the initial state, but not the correction term.

Theorem 5.4 *Complete N-step observability at time t_0 is neither a necessary nor a sufficient condition for a unique solution of Problem $\mathcal{A}_{CT}$.*

Proof

The result will be proved using simple counter-examples in the case $n = m = 2$, $p = 1$, and $N = 3$ and for $k = 0, 1, 2$.

We consider the system (5.3) with

$$A_0 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \qquad A_1 = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}, \tag{5.41}$$

and $B_k = I$ for $k = 0, 1$.

To show that complete N-step observability is not a necessary condition for a unique solution of Problem $\mathcal{A}_{CT}$, we give an example in which

$$\text{Rank}(\mathcal{O}_3) < 2, \qquad \text{Rank}(\mathcal{T}_3) = 2. \tag{5.42}$$

We suppose that the data set is given by (5.4) with

$$C_0 = (0, 1), \qquad C_1 = (0, 1), \qquad C_2 = (1, 1). \tag{5.43}$$

In this case

$$\mathcal{O}_3 = \begin{pmatrix} C_0 \\ C_1 A_0 \\ C_2 A_1 A_0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \\ 0 & 2 \end{pmatrix}, \tag{5.44}$$

$$\mathcal{T}_3 = \begin{pmatrix} 0 \\ C_1 \\ C_2 A_1 + C_2 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 3 \end{pmatrix}, \tag{5.45}$$

and so (5.42) holds.

To show that complete N-step observability is not a sufficient condition for a unique solution of Problem $\mathcal{A}_{CT}$, we give an example in which

$$\text{Rank}(\mathcal{O}_3) = 2, \qquad \text{Rank}(\mathcal{T}_3) < 2. \tag{5.46}$$

If the data set is now given by

$$C_0 = (0, 0), \qquad C_1 = (1, 3), \qquad C_2 = (1, 1), \tag{5.47}$$

we have

$$\mathcal{O}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 4 \\ 0 & 2 \end{pmatrix}, \qquad \mathcal{T}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 3 \\ 1 & 3 \end{pmatrix}, \tag{5.48}$$

and so (5.46) holds. □

We have, however, the following special case as an example where complete $(N-1)$-step observability is a necessary and sufficient condition for a unique solution of Problem $\mathcal{A}_{CT}$. We consider again the time invariant system (5.32),(5.33). The experiments we describe in Section 5.3 are for a time invariant system, and so Theorem 5.5 is relevant for this case.

Theorem 5.5 *For the time invariant system (5.32),(5.33) with $m = n$ and B non-singular, Problem $\mathcal{A}_{CT}$ has a unique solution if and only if the system (5.32),(5.33) is completely $(N-1)$-step observable.*

Proof

We need to show that $\text{Rank}(\mathcal{T}_N) = n$ if and only if $\text{Rank}(\mathcal{O}_{N-1}) = n$.
We let $\mathcal{T}' = \mathcal{T}_N B^{-1}$, and note that since B is nonsingular, $\text{Rank}(\mathcal{T}_N) = n$ if and only if $\text{Rank}(\mathcal{T}') = n$.
For the time invariant system, we have

$$\mathcal{O}_{N-1} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{N-2} \end{pmatrix}, \qquad \mathcal{T}' = \begin{pmatrix} 0 \\ C \\ C(A+I) \\ \vdots \\ C(A^{N-2} + A^{N-3} + .. + I) \end{pmatrix} \tag{5.49}$$

Since each row of $\mathcal{T}'$ can be written as a linear combination of rows of $\mathcal{O}_{N-1}$, we have that $\text{Rank}(\mathcal{T}') \leq \text{Rank}(\mathcal{O}_{N-1})$. Further, since each row of $\mathcal{O}_{N-1}$ can be written as a linear combination of the rows of $\mathcal{T}'$, we have that $\text{Rank}(\mathcal{O}_{N-1}) \leq \text{Rank}(\mathcal{T}')$. Hence $\text{Rank}(\mathcal{O}_{N-1}) = \text{Rank}(\mathcal{T}') = n$ if and only if $\text{Rank}(\mathcal{T}_N) = n$, which proves the result. □

Finally, we show that a unique solution to Problem $\mathcal{A}_{CT}$ can be guaranteed provided we add a background term to the cost function. In this case, Problem $\mathcal{A}_{CT}$ is modified to

Problem $\mathcal{B}_{CT}$

Minimize with respect to $\mathbf{x}_1, .., \mathbf{x}_N; \mathbf{e}$

$$\mathcal{J} = \frac{1}{2}(\mathbf{e} - \mathbf{e}^b)^T Q^{-1}(\mathbf{e} - \mathbf{e}^b) + \frac{1}{2}\sum_{j=0}^{N-1}(C_j\mathbf{x}_j - \mathbf{y}_j)^T R_j^{-1}(C_j\mathbf{x}_j - \mathbf{y}_j) \tag{5.50}$$

subject to (5.3) with $\mathbf{x}_0$ *fixed.*

In (5.50), $\mathbf{e}^b \in \mathbb{R}^m$ is a background estimate for $\mathbf{e}$, and $Q^{-1} \in \mathbb{R}^{m\times m}$ is a symmetric, positive definite matrix, ideally representing the inverse covariance matrix of $(\mathbf{e} - \mathbf{e}^b)$.

Although it is known in the data assimilation literature that a background term is needed in some cases to give uniqueness to Problem $\mathcal{A}_{IS}$, the applications of Problem $\mathcal{A}_{CT}$ (ie, applications of the correction term technique) mentioned in Section 5.1 did not use a background term for the control vector. Wergen [87] added an extra term to the cost function which acted as a background term which constrained just one of the three model fields to be close to zero.

In analogy to the working of the previous subsection, the Hessian of $\mathcal{J}$ with respect to $\mathbf{e}$ is $(Q^{-1} + \tilde{A}_{CT})$, and the following result holds.

Theorem 5.6 *Problem $\mathcal{B}_{CT}$ has a unique solution.*

Proof

The Hessian matrix $(Q^{-1} + \tilde{A}_{CT})$ is the sum of a positive definite and a positive semi-definite matrix, and so is positive definite. Hence Problem $\mathcal{B}_{CT}$ has a unique solution. □

It could also be deduced from the results in Bennett and Miller's paper [8] on the importance of a background estimate of the initial state, that when terms representing model error are estimated, a background estimate for these terms is sufficient for uniqueness in the linear case.

5.2.3 Using both the initial state and the correction term as control vectors

We notice that the system (5.3),(5.4) can equivalently be written as

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + B_k\mathbf{e}_k, \tag{5.51}$$

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \qquad k = 0,..,N-1, \tag{5.52}$$

with observations

$$\mathbf{y}_k = C_k\mathbf{x}_k^t + \boldsymbol{\delta}_k, \qquad k = 0,..,N-1. \tag{5.53}$$

It is helpful to rewrite this system as the *augmented system*

$$\mathbf{w}_{k+1} = M_k\mathbf{w}_k, \tag{5.54}$$

$$\mathbf{y}_k = \tilde{C}_k\mathbf{w}_k^t + \boldsymbol{\delta}_k, \tag{5.55}$$

where $\mathbf{w}_k \in \mathbb{R}^{n+m}$, $M_k \in \mathbb{R}^{(n+m)\times(n+m)}$ and $\tilde{C}_k \in \mathbb{R}^{p_k\times(n+m)}$ are given by

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{e}_k \end{pmatrix}, \qquad M_k = \begin{pmatrix} A_k & B_k \\ 0 & I \end{pmatrix}, \qquad \tilde{C}_k = \begin{pmatrix} C_k & 0 \end{pmatrix}. \tag{5.56}$$

In this augmented system, $\mathbf{w}_k$ is the *augmented state vector* and $\mathbf{w}_0$ is the *augmented control vector*. The augmented state transition matrix is $\tilde{\Phi}(k,j)$, and we have

$$\tilde{\Phi}(k,0) = \tilde{\Phi}_k = \begin{pmatrix} \Phi_k & \tilde{B}_k \\ 0 & I \end{pmatrix}. \tag{5.57}$$

Problem $\mathcal{A}_{ISCT}$ can now equivalently be written

Problem $\mathcal{A}_{ISCT}$

Minimize with respect to $\mathbf{x}_0,..,\mathbf{x}_N;\mathbf{e}$

$$\mathcal{J} = \frac{1}{2}\sum_{j=0}^{N-1}(\tilde{C}_j\mathbf{w}_j - \mathbf{y}_j)^T R_j^{-1}(\tilde{C}_j\mathbf{w}_j - \mathbf{y}_j) \tag{5.58}$$

subject to (5.54).

In this form, we see that Problem $\mathcal{A}_{ISCT}$ is just Problem $\mathcal{A}_{IS}$ applied to the system (5.54), and so the theory of Section 5.2.1 applies here. From Theorem 5.1 we

know that Problem $\mathcal{A}_{ISCT}$ has a unique minimum if and only if the N-step observability matrix $\tilde{\mathcal{O}}_N$ has rank $(n+m)$, where

$$\tilde{\mathcal{O}}_N = \begin{pmatrix} \tilde{C}_0 \\ \tilde{C}_1\tilde{\Phi}_1 \\ \vdots \\ \tilde{C}_{N-1}\tilde{\Phi}_{N-1} \end{pmatrix}. \tag{5.59}$$

We now consider observability of the augmented system (5.54),(5.55) in terms of observability of the original system (5.3),(5.4). Observability of the augmented system concerns the ability to determine $\mathbf{w}_0$ or equivalently $\mathbf{x}_0$ and $\mathbf{e}$ from the observations, and observability of the original system concerns the ability to reconstruct $\mathbf{x}_0$ from the same observations. We now show that the following result holds.

Theorem 5.7 *Necessary conditions for Problem $\mathcal{A}_{ISCT}$ to have a unique solution are that the original system (5.3),(5.4) is completely N-step observable at time t_0 and that Rank$(\mathcal{T}_N) = m$.*

Proof

From Theorem 5.1, Problem $\mathcal{A}_{ISCT}$ has a unique solution if and only if Rank$(\tilde{\mathcal{O}}_N) = (n+m)$, with $\tilde{\mathcal{O}}_N$ given in (5.59).

Since

$$\tilde{C}_k\tilde{\Phi}_k = (C_k\Phi_k, C_k\tilde{B}_k), \qquad k = 0, .., N-1 \tag{5.60}$$

we have

$$\tilde{\mathcal{O}}_N = (\mathcal{O}_N, \mathcal{T}_N). \tag{5.61}$$

Let $\mathbf{v} \in \mathbb{R}^{n+m}$ be arbitrary, and suppose

$$\tilde{\mathcal{O}}_N\mathbf{v} = \mathcal{O}_N\mathbf{v}_1 + \mathcal{T}_N\mathbf{v}_2, \tag{5.62}$$

with $\mathbf{v}_1 \in \mathbb{R}^n$ and $\mathbf{v}_2 \in \mathbb{R}^m$.

If Rank$(\mathcal{O}_N) < n$, then there exists a non-zero vector $\mathbf{v}_1'$ such that $\mathcal{O}_N\mathbf{v}_1' = 0$. Hence with this choice of $\mathbf{v}_1$ and with $\mathbf{v}_2 = 0$, there exists a non-zero vector $\mathbf{v}$ such that $\tilde{\mathcal{O}}_N\mathbf{v} = 0$.

Hence, Rank$(\mathcal{O}_N) = n$, or equivalently by Theorem 2.2, complete N-step observability is a necessary condition for a unique solution of Problem $\mathcal{A}_{ISCT}$.

By a similar argument, we have that Rank$(\mathcal{T}_N) = m$ is also a necessary condition for a unique solution of Problem $\mathcal{A}_{ISCT}$. □

One simple example for which Problem $\mathcal{A}_{ISCT}$ has a unique solution is the case where C_0 and C_1 both have rank n, (as is the case in our experiments where we use the full set of observations), as we now show.

We suppose $\mathbf{v} \in \mathbb{R}^{n+m}$ is arbitrary and that $\tilde{\mathcal{O}}_N \mathbf{v} = 0$. Then by (5.62)

$$\mathcal{O}_N \mathbf{v}_1 + \mathcal{T}_N \mathbf{v}_2 = 0, \tag{5.63}$$

and by equation (5.60)

$$C_k \Phi_k \mathbf{v}_1 + C_k \tilde{B}_k \mathbf{v}_2 = 0, \qquad k = 0, .., N-1. \tag{5.64}$$

With $k = 0$ we have

$$C_0 \mathbf{v}_1 = 0, \tag{5.65}$$

and hence $\mathbf{v}_1 = 0$ since C_0 has rank n. With $\mathbf{v}_1 = 0$ in the equation for $k = 1$ we have

$$C_1 \tilde{B}_1 \mathbf{v}_2 = C_1 B_0 \mathbf{v}_2 = 0, \tag{5.66}$$

and since B_0 has rank m and C_1 has rank n, we also have $\mathbf{v}_2 = 0$. We have shown that in this case,

$$\tilde{\mathcal{O}}_N \mathbf{v} = 0 \Rightarrow \mathbf{v} = 0, \tag{5.67}$$

so in this case the augmented system is completely N-step observable, and hence Problem $\mathcal{A}_{ISCT}$ has a unique solution.

The necessary conditions given in Theorem 5.7 are not in general sufficient for a unique solution of Problem $\mathcal{A}_{ISCT}$. We show this for the linear, time-invariant system (5.32), (5.33), which can equivalently be written as the augmented system

$$\mathbf{w}_{k+1} = M \mathbf{w}_k, \tag{5.68}$$

$$\mathbf{y}_k = \tilde{C} \mathbf{w}_k^t + \boldsymbol{\delta}_k, \tag{5.69}$$

where

$$M = \begin{pmatrix} A & B \\ 0 & I \end{pmatrix}, \qquad \tilde{C} = \begin{pmatrix} C & 0 \end{pmatrix}. \tag{5.70}$$

The following result is also applicable to the system we use in the experiments we describe in Section 5.3, and we use it later.

Theorem 5.8 *For the time invariant system (5.68), (5.69) with $m = n$ and B nonsingular, Problem $\mathcal{A}_{ISCT}$ has a unique solution if and only if Rank$(C) = n$.*

Proof

By Theorem 5.1, Problem $\mathcal{A}_{ISCT}$ has a unique solution if and only if the augmented system (5.68), (5.69) is completely N-step observable. We show that if Rank$(C) < n$, the system (5.68), (5.69) is not completely observable, and hence not completely ν-step observable for any ν by Theorem 2.3 Part a.

By the negation of the Hautus condition (Theorem 2.4), the system is not completely observable if there exists a non-zero vector $\mathbf{v} \in \mathbb{R}^{2n}$ and $\lambda \in \mathbb{C}$ such that

$$(M - \lambda I)\mathbf{v} = 0, \tag{5.71}$$

$$\tilde{C}\mathbf{v} = 0, \tag{5.72}$$

or equivalently

$$(A - \lambda I)\mathbf{v}_1 + B\mathbf{v}_2 = 0, \tag{5.73}$$

$$(I - \lambda I)\mathbf{v}_2 = 0, \tag{5.74}$$

$$C\mathbf{v}_1 = 0, \tag{5.75}$$

where $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^n$.

Suppose that Rank$(C) < n$. Then there exists a non-zero $\mathbf{v}_1$ such that $C\mathbf{v}_1 = 0$. Let $\mathbf{v}_2$ be given by

$$\mathbf{v}_2 = B^{-1}(A - \lambda I)\mathbf{v}_1, \tag{5.76}$$

and $\lambda = 1$. Then (5.73)– (5.75) or equivalently (5.71), (5.72) hold for a non-zero vector $\mathbf{v}$, and $\lambda = 1$. Hence, if Rank$(C) < n$ the augmented system is not completely observable, and Problem $\mathcal{A}_{ISCT}$ does not have a unique solution.

The fact that Problem $\mathcal{A}_{ISCT}$ has a unique solution if $\text{Rank}(C) = n$ follows from our previous remarks which showed that this is true for the general time-varying system with $\text{Rank}(C_0) = n$, $\text{Rank}(C_1) = n$. □

By Theorem 5.2, we know that we can guarantee a unique solution to Problem $\mathcal{A}_{ISCT}$ by adding a background estimate of the augmented control vector $\mathbf{w}_0$ to the cost function, and so we formulate the following problem.

Problem $\mathcal{B}_{ISCT}$

Minimize with respect to $\mathbf{w}_0, .., \mathbf{w}_N$

$$\mathcal{J} = \frac{1}{2}(\mathbf{w}_0 - \mathbf{w}_0^b)^T \tilde{P}_0^{-1}(\mathbf{w}_0 - \mathbf{w}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\tilde{C}\mathbf{w}_j - \mathbf{y}_j)^T R_j^{-1}(\tilde{C}\mathbf{w}_j - \mathbf{y}_j) \qquad (5.77)$$

subject to (5.54).

In (5.77), $\mathbf{w}_0^b \in \mathbb{R}^{n+m}$ is a background estimate of $\mathbf{w}_0$, and $\tilde{P}_0^{-1} \in \mathbb{R}^{(n+m)\times(n+m)}$ is symmetric, positive definite. By Theorem 5.2, Problem $\mathcal{B}_{ISCT}$ has a unique solution.

We now show that if the original time-varying system (5.3), (5.4) is completely N-step observable, and if we use a background estimate of $\mathbf{e}$ only, we are again guaranteed a unique solution. We formulate the following modification of Problem $\mathcal{A}_{ISCT}$.

Problem $\mathcal{C}_{ISCT}$

Minimize with respect to $\mathbf{w}_0, .., \mathbf{w}_N$

$$\mathcal{J} = \frac{1}{2}(\mathbf{e}_0 - \mathbf{e}^b)^T Q^{-1}(\mathbf{e}_0 - \mathbf{e}^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\tilde{C}\mathbf{w}_j - \mathbf{y}_j)^T R_j^{-1}(\tilde{C}\mathbf{w}_j - \mathbf{y}_j) \qquad (5.78)$$

subject to (5.54).

In (5.78), $\mathbf{e}^b$ and Q^{-1} are as defined in Problem $\mathcal{B}_{CT}$. We now give conditions under which Problem $\mathcal{C}_{ISCT}$ has a unique solution.

Theorem 5.9 *Problem $\mathcal{C}_{ISCT}$ has a unique solution if and only if the original system (5.3),(5.4) is completely N-step observable at time t_0.*

Proof

We note that (5.78) can equivalently be written

$$\mathcal{J} = \frac{1}{2}\sum_{j=0}^{N-1}(D_j\mathbf{w}_j - \mathbf{z}_j)^T S_j^{-1}(D_j\mathbf{w}_j - \mathbf{z}_j), \tag{5.79}$$

where $D_j \in \mathbb{R}^{(p_j+m)\times(n+m)}$, $S_j^{-1} \in \mathbb{R}^{(p_j+m)\times(p_j+m)}$ and $\mathbf{z}_j \in \mathbb{R}^{p_j+m}$ are given by

$$D_j = \begin{pmatrix} C_j & 0 \\ 0 & I \end{pmatrix}, \qquad S_j^{-1} = \begin{pmatrix} R_j^{-1} & 0 \\ 0 & \frac{1}{N}Q^{-1} \end{pmatrix}, \qquad \mathbf{z}_j = \begin{pmatrix} \mathbf{y}_j \\ \mathbf{e}^b \end{pmatrix}. \tag{5.80}$$

Since the matrices S_k^{-1} are symmetric, positive definite, we can apply Theorem 5.1 to see that Problem $\mathcal{C}_{ISCT}$ has a unique solution if and only if the observability matrix $\hat{\mathcal{O}}_N$ has rank $(n+m)$, where

$$\hat{\mathcal{O}}_N = \begin{pmatrix} D_0\tilde{\Phi}_0 \\ D_1\tilde{\Phi}_1 \\ \vdots \\ D_{N-1}\tilde{\Phi}_{N-1} \end{pmatrix}. \tag{5.81}$$

It therefore suffices to show that $\text{Rank}(\hat{\mathcal{O}}_N) = (n+m)$ if and only if $\text{Rank}(\mathcal{O}_N) = n$. We note that

$$D_k\tilde{\Phi}_k = \begin{pmatrix} C_k & 0 \\ 0 & I \end{pmatrix}\begin{pmatrix} \Phi_k & \tilde{B}_k \\ 0 & I \end{pmatrix} = \begin{pmatrix} C_k\Phi_k & C_k\tilde{B}_k \\ 0 & I \end{pmatrix}. \tag{5.82}$$

Let $\mathbf{v} \in \mathbb{R}^{n+m}$ be arbitrary with $\mathbf{v} = \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix}$, $\mathbf{v}_1 \in \mathbb{R}^n$, and $\mathbf{v}_2 \in \mathbb{R}^m$. We have

$$\hat{\mathcal{O}}_N\mathbf{v} = 0 \quad \text{if and only if} \quad D_k\tilde{\Phi}_k\mathbf{v} = 0 \qquad k = 0,..,N-1, \tag{5.83}$$

which holds if and only if

$$C_k\Phi_k\mathbf{v}_1 + C_k\tilde{B}_k\mathbf{v}_2 = 0, \tag{5.84}$$

$$\mathbf{v}_2 = 0, \qquad k = 0,..,N-1, \tag{5.85}$$

ie, if and only if

$$C_k\Phi_k\mathbf{v}_1 = 0 \qquad k = 0,..,N-1, \tag{5.86}$$

or equivalently

$$\mathcal{O}_N \mathbf{v}_1 = 0. \tag{5.87}$$

Hence, there exists a non-zero vector $\mathbf{v} \in \mathbb{R}^{n+m}$ such that $\hat{\mathcal{O}}_N \mathbf{v} = 0$ if and only if there exists a non-zero vector $\mathbf{v}_1 \in \mathbb{R}^n$ such that $\mathcal{O}_N \mathbf{v}_1 = 0$. It follows that $\hat{\mathcal{O}}_N$ has rank $(n+m)$ if and only if $\mathcal{O}_N$ has rank n, which proves the result. □

In Section 5.5, we present a summary of the theoretical results in this chapter. We now compare the performance of the initial state, correction term and both together as control vectors practically, in experiments with a simple model.

5.3 Description of the experiments

The aim of these experiments is to compare the performance of variational assimilation using the different control vectors: the initial state, the correction term and the augmented control vector containing the initial state and the correction term. This is done for a "perfect model" with unknown initial state, and for an "imperfect model" in which the source term is unknown. We also investigate the impact of p, the number of observations available at each timestep, on the results, and the impact of including a background term in the cost function, constraining the correction term to be small. Finally, we try using a correction term of dimension m less than the dimension n, which corrects just an area close to the source term.

5.3.1 The model and observations

The model

In these experiments, we use the heat equation model with a source term on the time interval $[0, 1]$, which we described in Chapter 3, Section 3.5, given by

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{s}, \qquad k = 0, .., N-1. \tag{5.88}$$

We take $N = 80$, $T = 1$, $J = 16$ and $\sigma = 0.1$, so $\Delta t = \frac{1}{80}$, $\Delta z = \frac{1}{16}$, $n = 15$ and $\mu = 0.32$.

The true model state

We suppose that the true model state $\mathbf{x}_k^t$ satisfies (5.88) started from the true initial state $\mathbf{x}_0^t$ is given by

$$(x_j^0)^t = 1, \qquad j = 1, .., n. \tag{5.89}$$

Model error

As a source of model error, we suppose that the constant source term is omitted from the model equations. Hence, in the "imperfect" model, $\mathbf{s}$ is set to zero.

Observations

We suppose that we have error free observations at p of the 15 grid points at every timestep on the interval $[0, 0.5]$, ie for $\frac{N}{2} = 40$ timesteps, and that after this no further observations are available. Hence, the observations are given by

$$\mathbf{y}_k = C\mathbf{x}_k^t, \qquad k = 0, ..., \frac{N}{2} - 1, \tag{5.90}$$

where the observational matrix $C \in \mathbb{R}^{p \times n}$ has a simple form since the observation positions coincide with the grid points. The positions of the observations used in each case are shown in the figures.

5.3.2 The minimization problem

Our aim is to estimate the true model state $\mathbf{x}_k^t$ from the observations (5.90) using the model

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{s} + B\mathbf{e}, \qquad k = 0, .., \frac{N}{2} - 1, \tag{5.91}$$

where $\mathbf{e} \in \mathbb{R}^m$ is the correction term and $B \in \mathbb{R}^{n \times m}$ is the identity matrix if $m = n$, and if $m < n$, then B is a transformation matrix which limits the effect of the correction term to a limited area of the model domain.

We minimize the cost function

$$\mathcal{J} = \frac{1}{2}\mathbf{e}^T Q^{-1}\mathbf{e} + \frac{1}{2}\sum_{j=0}^{\frac{N}{2}-1}(C\mathbf{x}_j - \mathbf{y}_j)^T R^{-1}(C\mathbf{x}_j - \mathbf{y}_j), \tag{5.92}$$

subject to (5.91), where $R^{-1} = \frac{2}{N} I \in \mathbb{R}^{p \times p}$, and $Q^{-1} = qI \in \mathbb{R}^{n \times n}$. The matrices R^{-1} give equal weight to all observations, and are not supposed to represent error covariances. The value of q is sometimes taken to be zero, in which case we do not constrain the size of the correction term to be small.

The adjoint model is

$$\boldsymbol{\lambda}_k = A^T \boldsymbol{\lambda}_{k+1} - C^T R^{-1}(C\mathbf{x}_k - \mathbf{y}_k), \qquad k = \frac{N}{2} - 1, .., 0, \tag{5.93}$$

with

$$\boldsymbol{\lambda}_{\frac{N}{2}} = 0. \tag{5.94}$$

The gradients of the Lagrange function $\mathcal{L}$ associated with $\mathcal{J}$ with respect to the control vectors are

$$\nabla_{\mathbf{x}_0} \mathcal{L} = -\boldsymbol{\lambda}_0, \tag{5.95}$$

$$\nabla_{\mathbf{e}} \mathcal{L} = Q^{-1}\mathbf{e} - B^T \sum_{j=1}^{\frac{N}{2}-1} \boldsymbol{\lambda}_j. \tag{5.96}$$

We verified that the system (5.90),(5.91) is completely observable even when just one observation is used, using the Matlab package for calculating matrix rank. Hence, we are guaranteed a unique solution even when both control vectors are used provided $q > 0$, by Theorem 5.9.

The minimization algorithm used in these experiments is the conjugate gradient method (CGM). In Chapter 2, Section 2.4 we outlined this method for an unconstrained minimization problem. In Subsection 5.3.3 below we explain how we implement the CGM for our constrained minimization problem.

5.3.3 The CGM for a constrained minimization problem

We aim to solve the constrained minimization problem Problem $\mathcal{A}_{ISCT}$ or one of its variants. As we saw in Section 5.2, this can be written as an unconstrained minimization problem explicitly in terms of the control vector $\mathbf{u}$, in the following form

$$\tilde{\mathcal{J}} = \frac{1}{2}\mathbf{u}^T \tilde{A}\mathbf{u} + \tilde{\mathbf{b}}^T\mathbf{u} + \tilde{c}, \tag{5.97}$$

where the control vector $\mathbf{u}$ might be the initial state, correction term or the augmented vector consisting of the initial state and the correction term. Unless stated otherwise, the stopping criterion used in the conjugate gradient descent is

$$\|\nabla_{\mathbf{u}}\tilde{\mathcal{J}}\| < 10^{-6}. \tag{5.98}$$

If this stopping criterion is not satisfied within 100 iterations, the descent process is terminated anyway.

For this unconstrained minimization problem, we use the conjugate gradient method (CGM) outlined in Chapter 2, Section 2.4. The CGM algorithm can be written in terms of $\mathbf{u}^k$, the kth iterate of the control vector, as summarized below, where $< .,. >$ represents the Euclidian inner product in $\mathbb{R}^n$:

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \rho^k \mathbf{d}^k, \tag{5.99}$$

where

$$\mathbf{d}^{k+1} = -\mathbf{r}^{k+1} + \beta^k \mathbf{d}^k, \tag{5.100}$$

$$\mathbf{r}^k = \nabla_{\mathbf{u}}\tilde{\mathcal{J}} = \tilde{A}\mathbf{u}^k + \tilde{\mathbf{b}}, \tag{5.101}$$

with

$$\rho^k = \frac{< \mathbf{r}^k, \mathbf{d}^k >}{< \mathbf{d}^k, \tilde{A}\mathbf{d}^k >} \qquad \beta^k = \frac{< \mathbf{r}^{k+1}, \tilde{A}\mathbf{d}^k >}{< \mathbf{d}^k, \tilde{A}\mathbf{d}^k >} \tag{5.102}$$

and with

$$\mathbf{d}^0 = -\mathbf{r}^0. \tag{5.103}$$

In practice we do not have an explicit form of the Hessian matrix $\tilde{A}$. We must therefore find a way of implementing (5.99)-(5.102) which does not use explicit knowledge of the Hessian. The Hessian $\tilde{A}$ appears in the expressions for the residual $\mathbf{r}^k$, and in the inner products $< \mathbf{r}^{k+1}, \tilde{A}\mathbf{d}^k >$ and $< \mathbf{d}^k, \tilde{A}\mathbf{d}^k >$. We now discuss how these terms may be evaluated without explicit knowledge of $\tilde{A}$.

Calculating the residuals $\mathbf{r}^k$

Firstly, we notice that the residual $\mathbf{r}^k$ is the gradient of $\tilde{\mathcal{J}}$ with respect to the control variable $\mathbf{u}^k$. The gradient of $\tilde{\mathcal{J}}$ with respect to $\mathbf{u}^k$ is the same as the gradient of the Lagrangian $\mathcal{L}$ with respect to $\mathbf{u}^k$ as defined in (5.95),(5.96).

Calculating $< \mathbf{r}^{k+1}, \tilde{A}\mathbf{d}^k >$

From (5.99) and (5.101) we have

$$\tilde{A}\mathbf{d}^k = (\mathbf{r}^k - \mathbf{r}^{k+1})/\rho^k. \tag{5.104}$$

Everything on the right hand side of (5.104) is known by the time we need to evaluate $< \mathbf{r}^{k+1}, \tilde{A}\mathbf{d}^k >$, so $\tilde{A}\mathbf{d}^k$ can be evaluated easily.

Calculating $< \mathbf{d}^k, \tilde{A}\mathbf{d}^k >$

This expression can be built up using the following iteration. Starting from

$$\boldsymbol{\xi}_0 = \mathbf{d}^k, \tag{5.105}$$

$$\boldsymbol{\zeta}_0 = R^{-1}C\boldsymbol{\xi}_0, \tag{5.106}$$

$$\boldsymbol{\sigma}_0 = \boldsymbol{\zeta}_0^T\boldsymbol{\zeta}_0, \tag{5.107}$$

for $i = 1, .., \frac{N}{2} - 1$ we let

$$\boldsymbol{\xi}_i = \boldsymbol{\psi}_i(\boldsymbol{\xi}_{i-1}), \tag{5.108}$$

$$\boldsymbol{\zeta}_i = R^{-1}C\boldsymbol{\xi}_i, \tag{5.109}$$

$$\boldsymbol{\sigma}_i = \boldsymbol{\sigma}_{i-1} + \boldsymbol{\zeta}_i^T\boldsymbol{\zeta}_i, \tag{5.110}$$

where

$$\boldsymbol{\psi}_i(\boldsymbol{\xi}_{i-1}) = A\boldsymbol{\xi}_{i-1} \quad \text{if} \quad \mathbf{u} = \mathbf{x}_0, \tag{5.111}$$

$$\boldsymbol{\psi}_i(\boldsymbol{\xi}_{i-1}) = A\boldsymbol{\xi}_{i-1} + B\boldsymbol{\xi}_0 \quad \text{if} \quad \mathbf{u} = \mathbf{e}, \tag{5.112}$$

$$\boldsymbol{\psi}_i(\boldsymbol{\xi}_{i-1}) = \begin{pmatrix} A & B \\ 0 & I \end{pmatrix} \boldsymbol{\xi}_{i-1} \quad \text{if} \quad \mathbf{u} = \begin{pmatrix} \mathbf{x}_0 \\ \mathbf{e} \end{pmatrix}. \tag{5.113}$$

Then we have

$$< \mathbf{d}^k, \tilde{A}\mathbf{d}^k >= \boldsymbol{\sigma}_{\frac{N}{2}-1}. \tag{5.114}$$

5.3.4 The experiments

We investigate the performance of data assimilation using the initial state, the correction term and both together as control vectors for each of the following cases.

Case a: Perfect model, unknown initial state

In this case, the first guess of the initial state is

$$x_j^0 = 2, \qquad j = 1, .., n, \tag{5.115}$$

rather than (5.89).

Case b: Imperfect model, known initial state

The source term **s** is assumed to be zero in this imperfect model, but this time the true initial state is known.

Case c: Imperfect model, unknown initial state

Here we use the imperfect model of Case **b** with the first guess initial state specified in Case **a**.

Data assimilation is carried out over the time interval $[0, 0.5]$, where observations are available. We then suppose that no more observations are available, and see whether any benefits of the assimilation are maintained in a model run continued on the time interval $[0.5, 1]$. This type of experiment in allows us to ascertain whether the assimilation results in an improved "forecast", which is the ultimate aim of operational applications of data assimilation [87]. This is especially important when we are testing the correction term technique, because as Wergen found [87], an improved solution during the assimilation interval does not always give an improved forecast using this technique.

The simplicity of the linear, time-invariant model, with small state dimension, the fact that it is completely observable for all values of p and the fact that the observations contain no noise and that the model error we examine is constant in time means, of course, that the situation we examine here is very different to that of operational NWP. Further, the fact that the model is strongly dissipative means that some aspects of the results will not hold in general, as we will point out.

However, this setup allows us to examine the relative efficiency of the various control vectors in using the observational data to correct for wrong initial conditions

and for model error which is constant in time. The results from these experiments should enable us to make conclusions which will apply more generally.

5.4 Results

The figures referred to in the text may be found at the end of this section.

5.4.1 Experiments using the initial state as the control vector

We carry out data assimilation using the initial state as the control vector for a perfect model with unknown initial state, and for an imperfect model with known initial state. Here $\mathbf{e}$ is set fixed at zero.

Experiment 1a: Perfect model, unknown initial state

From the theory of Section 5.2.1 we know that, since the system is completely observable and we have perfect model and perfect observations, it should be possible to reconstruct perfectly the true initial state and hence the true solution for subsequent times from the observations.

Fig. 5.1 shows that with 5 observations ($p = 5$), the true initial state is reconstructed exactly. This requires 20 iterations of the minimization algorithm. The results at $t = 0.5$, at the end of the assimilation interval, match the true results exactly, and so, since the model is perfect, the "forecast" started at $t = 0.5$ still matches the true solution at $t = 1$.

If fewer observations are used, fewer iterations of the descent algorithm are needed to satisfy the stopping criterion. Fig. 5.2 shows the results for $p = 1$. Here the match to the true solution at the initial time is very poor, but at subsequent times is good, and the forecast initiated at $t = 0.5$ matches the true solution exactly. This illustrates that since the model is strongly dissipative, the solution is not very sensitive to the initial state. Even using a stricter stopping criterion, it is not possible to obtain a more accurate result in the case $p = 1$, because of numerical round-off error.

Experiment 1b: Imperfect model, known initial state

In this case we still expect that the minimization problem has a unique solution, but the optimal initial state found will not be the true one, but that which gives the solution of the imperfect model which is closest to the observations.

The true initial state is known, and the minimization iterations are started from this value. If a different start guess for the initial state is used, the same results are found, but this generally takes a couple of iterations more. Fig. 5.3 shows that when the full set of observations are used ($p = 15$), a smooth initial state is found with a higher value at the position of the source point. If fewer observations are used (Fig. 5.4 shows the results when $p = 5$), the initial state obtained is no longer smooth, but matches the true solution at the observation positions. However, the model dissipation soon acts to smooth out the solution.

The best results in each case are in the middle of the assimilation interval, at $t = 0.25$. This tendency of the variational assimilation method to give a closest fit to observations in the middle of the assimilation interval has often been noted [19]. In this case of an imperfect model, the method does not produce the true initial state but finds one for which the ensuing solution throughout the assimilation interval is closer to the observations. In this way, the effects of model error, rather than building up in time, have been spread throughout the assimilation interval, as noted in [87]. Although the assimilation has improved the results at $t = 0.5$, the benefits at $t = 1$ are much smaller, because the forecast has been carried out with an uncorrected imperfect model.

5.4.2 Experiments using the correction term as the control vector

Here we carry out data assimilation using the correction term as a control vector, and with $\mathbf{x}_0$ fixed. Comparison of these results with those using the initial state as the control vector gives a comparison of the efficiency of the two control vectors in each situation. We also examine the use of the background term $\frac{1}{2}\mathbf{e}^T q I \mathbf{e}$ in the cost function, with different values of q. We look first at the situation of the imperfect

model with known initial state, since this problem is more naturally treated by using the correction term as a control vector.

Experiment 2b: Imperfect model, known initial state

Since the system is completely observable and $\frac{N}{2} > n+1$, we know that Problem $\mathcal{A}_{CT}$ has a unique solution. Since in Case **b** the initial state is known and model error is constant in time, this method should find a correction term which exactly represents the model error, and hence be able to reproduce the true solution.

Fig. 5.5 shows this to be so when $p = 5$ using $q = 0$. If the correction term found is used in the forecast started at the end of the assimilation interval ($t = 0.5$), the forecast using this corrected model matches the true solution exactly. The number of iterations taken in this case is 17, very similar to the number taken in Experiment 1a using the initial state as the control vector; and as in that case, the number of iterations decreases when fewer observations are used. When only one observation is used with $q = 0$, (Fig. 5.6), the results are still good.

Experiments were also carried out using the background term with $q = 1$. For a given value of p, fewer iterations were needed to satisfy the stopping criterion than using $q = 0$, but the results were very slightly less accurate.

Experiment 2a: Perfect model, unknown initial state

Again, we expect a unique solution, but not the true solution. The method will find the correction term for which the model started from a wrong initial state is as close as possible to the observations.

In these experiments the correction term is not included in the forecast following the assimilation. This is because the correction term is supposed to compensate for the errors in the initial conditions throughout the assimilation period, and since the model is perfect, an improved solution at the end of the assimilation interval (at $t = 0.5$) will give an improved forecast.

If $p = 15$, ie, the full set of observations are used, (Fig. 5.7), the solution is closest to the true solution about halfway through the assimilation interval. At the end of the assimilation interval, it is hard to judge whether the assimilation has

produced a better solution than the first guess, as one over-estimates and the other under-estimates the true solution, as is the case throughout the forecast.

The results using $q = 0$ are very poor if fewer than the full set of 15 observations are used, however. The results in these cases are very close to the true solution at the observation positions, but have large spikes where observations are missing, as Fig. 5.8 shows in the case $p = 5$. If more observations are used, the result is closer to the true solution in more places, but the spikes in the data voids are larger. In this case, the solution does not make sense since the correction term produces a solution close to the true solution only at the observation positions.

Using a background term with $q = 1$ improves the situation by getting rid of the large spikes, but the solution still is not smooth. This is shown in Fig. 5.9 for $p = 5$. As in Experiment 2b, introducing this background term reduces the number of iterations needed before the stopping criterion is reached, from 35 to 10 iterations in this case ($p = 5$). Using a stricter stopping criterion does not produce better results. Increasing the value of q to 10 gives a smoother solution in just 10 iterations, and a very smooth solution in just 6 iterations if q is increased to 50 (Fig. 5.10). In this last case, it happens that the solution at the end of the assimilation interval is very close to the true solution, and stays close to it in the forecast.

These results show that although a background term for the control vector is not necessary for uniqueness, it is very important in practice to constrain the correction term to be close to zero to obtain a smooth solution, rather than a solution which is merely close to the true solution at the observation positions. We note that the poor results shown in Fig. 5.8 would have shown a reduction in the cost function and in the rms errors of the forecast period. This means that using the criteria of some of the earlier work on the correction term with no background in the cost function, we might have concluded that these results represented an improvement on the results with no assimilation.

5.4.3 Experiments with both control vectors used together

We now carry out data assimilation using the augmented control vector consisting of the initial state and the correction term. We aim to find a way of doing this

which gives the benefits obtained using each of the control vectors separately, while keeping the extra cost at an acceptable level. As illustrated above, there are some situations in which using the initial state as the control vector, and other situations in which using the correction term as the control vector works particularly well. If we do not know before starting the assimilation which control vector is better for the situation, we would like to be able to use both together and obtain the same benefits as if the preferable control vector had been chosen. This is examined here by looking at Cases **a** and **b** using the augmented control vector. We then look at Case **c**, the more general situation of an imperfect model with unknown initial state, to see whether by using the augmented control vector we can obtain the true solution in this case.

From the theory of Section 5.2, we know that for our time invariant system, the minimization problem with no background term using both control vectors has a unique solution if and only if the full set of 15 observations are used (Theorem 5.8), but that since the system is completely observable for any number of observations, using $q > 0$ ensures uniqueness (Theorem 5.9).

Experiment 3a: Perfect model, unknown initial state

In this case we want results to be as the case where we use only the initial state as the control vector. As expected, when 15 observations are used, the results match the true solution exactly. However, this takes more iterations (27 iterations) than in the same case when only the initial state is used as the control vector (10 iterations).

When no background term is used, the results using fewer than 15 observations are rather like the poor results of Experiment 2a using only the correction term as the control vector and no background term ($q = 0$), although in this case the solution does not deviate so far from the true solution in data void areas. Using a background term ($q > 0$) solves this problem, however, and an exact match to the true solution is achieved using $q = 5$. With $p = 1$, exactly the same results are obtained as in the case where the initial state is the only control vector; the same rather inaccurate initial state is found, but an exact match to the true solution is obtained at later times. Using both control vectors together it is necessary to

perform many more iterations of the descent algorithm than using just one control vector. In this case, 90 iterations are used in the case $p = 5$, though fewer iterations are needed when fewer observations are used, (26 iterations for $p = 1$).

Increasing q reduces the number of iterations required, however. These results show that when both control vectors are used, the background term is essential for sensible results, and that an appropriate choice of q is important to save extra cost.

Experiment 3b: Imperfect model, known initial state

In this case, we want the results to be as in the case where only the correction term is used as the control vector. Again, an exact match to the true solution is achieved using 15 observations. In the light of the previous results and the theory, we might have expected that using fewer observations and no background term it would not be possible to obtain reasonable results. However, results in this case are as good as those obtained in the case where only the correction term is used as the control vector, apart from a very slight inaccuracy in the initial state. Although we are not guaranteed a unique solution in this case, the first guess of the initial state is correct, so from a first guess which is close to the true solution, the true solution is found. As above, the number of iterations needed is larger than in the case where only one control vector is used, although the increase is not so large, (27 iterations rather than 17 iterations using just the correction term in the case $p = 5$). This time, however, adding the background term with $q = 1$ significantly increases the number of iterations needed. Further increasing q in this case leads to a deterioration in the results, and does not decrease the number of iterations.

Experiment 3c: Imperfect model, unknown initial state

In the experiments without the background term ($q = 0$), the results are much as in Experiment 3a. A perfect match to the true solution is obtained using the full set of 15 observations, but if fewer observations are used, the solution is close to the true solution at the observation positions but not where observations are missing. The case for $p = 5$ is shown in Fig. 5.11. Using different starting guesses in this case gives different solutions, which demonstrates that the minimization problem does

not have a unique solution using $q = 0$ and $p < 15$.

Adding the background term (using $q > 0$), ensures uniqueness and results in good solutions, although for smaller values of p the results are not completely smooth at the beginning of the assimilation interval. Figs. 5.12 and 5.14 show the results for $p = 5$ and $p = 10$, respectively, with $q = 1$. In all cases the match to the true solution is very good at the end of the assimilation, and the forecast initiated at this time using the correction term found in the assimilation maintains a perfect match to the true solution.

Increasing the value of q to 10 gives a smoother solution for smaller values of p, and reduces the number of iterations needed. However, further increasing q to 50 leads to a less accurate match to the true solution as Fig. 5.13 shows in the case $p = 5$, and little further reduction in the number of iterations needed.

5.4.4 Reducing the dimension of the correction term vector

In this example, model error is due to the omission of a source term which is only nonzero at one gridpoint. We now suppose we know that model error is localized, and suppose that we know approximately this location. In this case we are able to reduce the dimension of the correction term to $m < n$ (provided the correction term is not also supposed to correct for errors in the initial state). When using $m < n$, we suppose that the correction term covers an area centred on the location of the source term.

Experiment 4b: Imperfect model, known initial state

Using values of $m < n$ improves the efficiency of the method. As before, using the correction term as the control vector it is possible to perfectly reconstruct the true solution from the observations using $p = 5$ and $q = 0$. If $m < n$, however, these results are achieved using fewer iterations, just 4 iterations for $m = 5$ and just 3 iterations for $m = 3$. Before, using $m = n = 15$, 17 iterations were required for the same results.

Using only one observation, the results were slightly more accurate using $m = 5$ than using $m = 15$, as comparing Fig. 5.16 with Fig. 5.15 illustrates. In this case, the observation is not in the area the correction term covers. The number of iterations required was reduced from 7 iterations for $m = n = 15$ to just 2 iterations for $m = 5$.

Experiment 4c: Imperfect model, unknown initial state

Here again reducing the dimension of the correction term improves the efficiency of the method. Fig. 5.17 shows the results using both control vectors and using $m = n = 15$ and 5 observations with $q = 1$. This requires 77 iterations of the descent algorithm. Fig. 5.18 shows that the results using $m = 3$ are just as good, and in this case only 31 iterations are required.

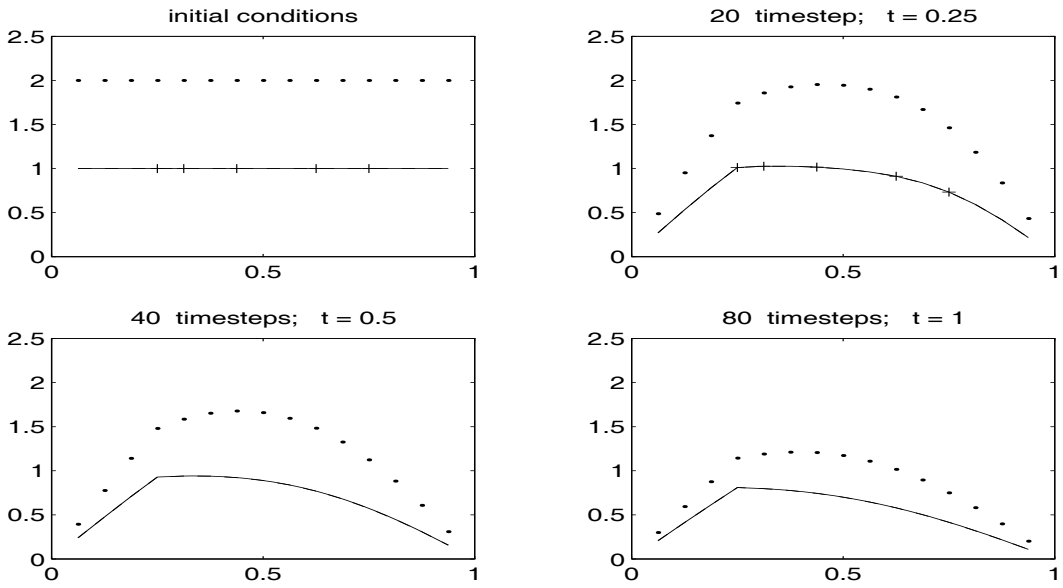

Figure 5.1: Variational assimilation using the initial state as the control vector. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 5 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

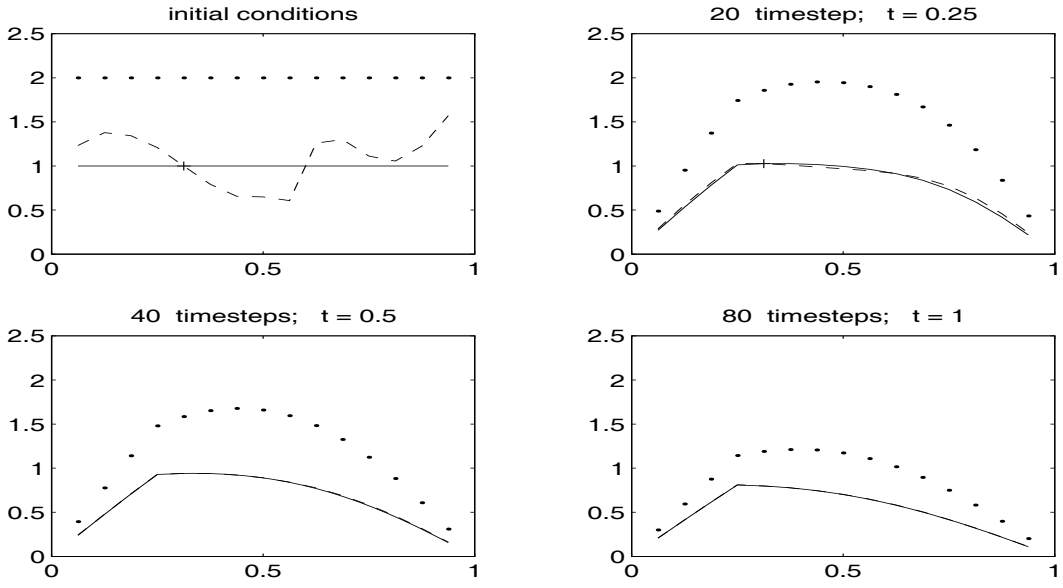

Figure 5.2: As Fig. 5.1, but using only 1 observation.

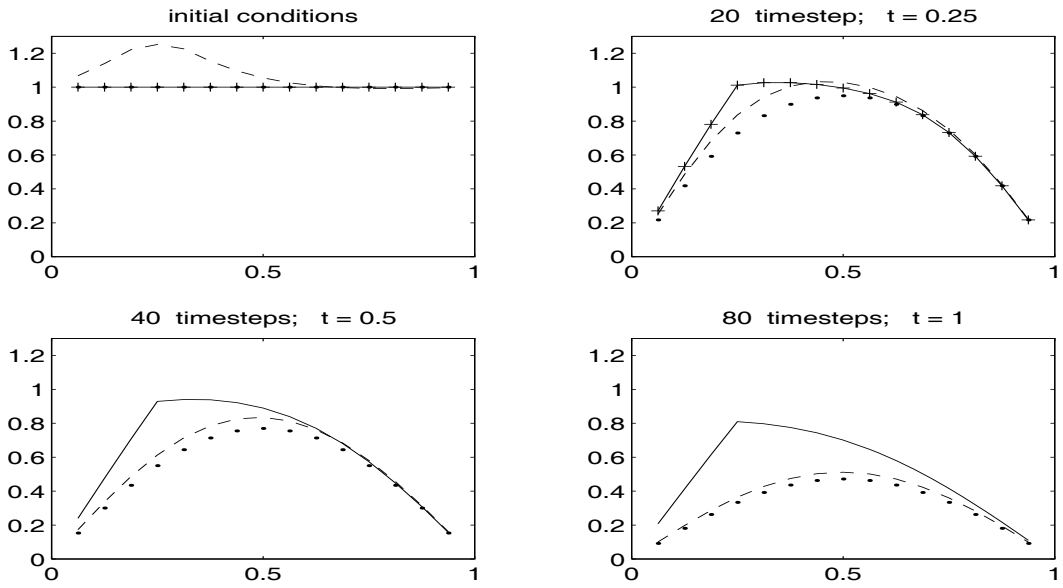

Figure 5.3: Variational assimilation using the initial state as the control vector. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 15 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

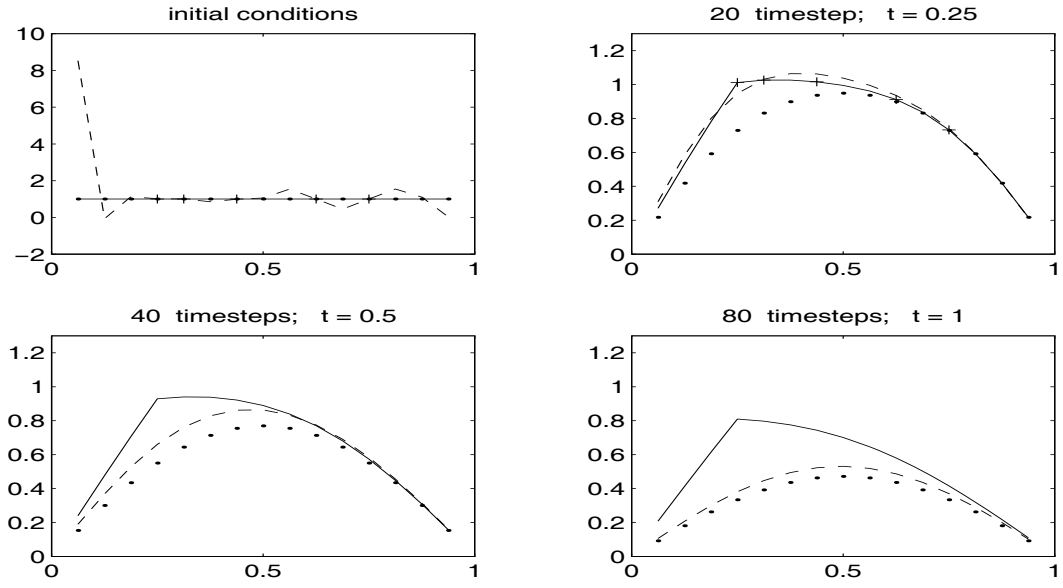

Figure 5.4: As Fig. 5.3, but using only 5 observations.

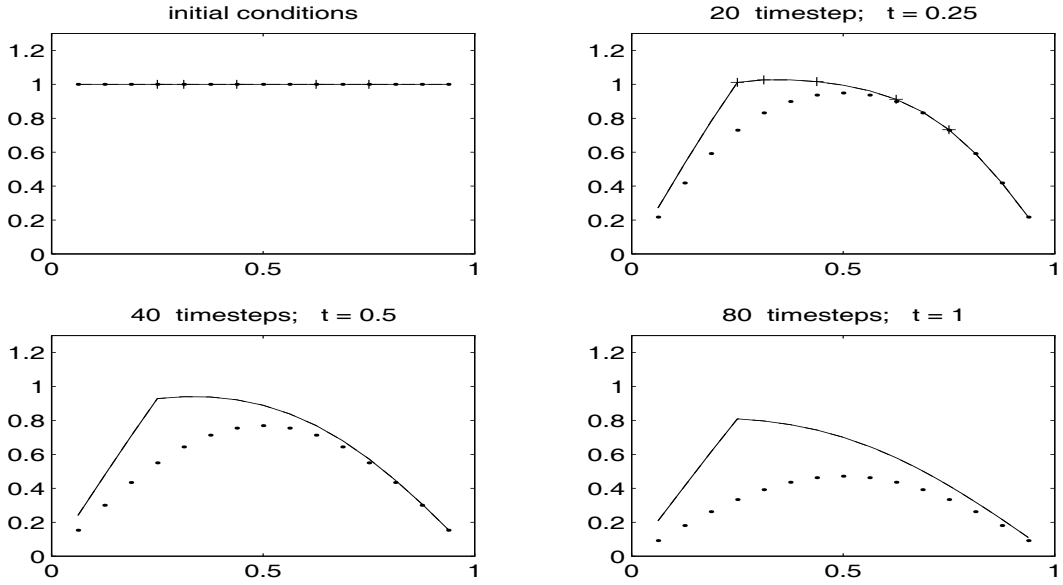

Figure 5.5: Variational assimilation using the correction term as the control vector, with $q = 0$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 5 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

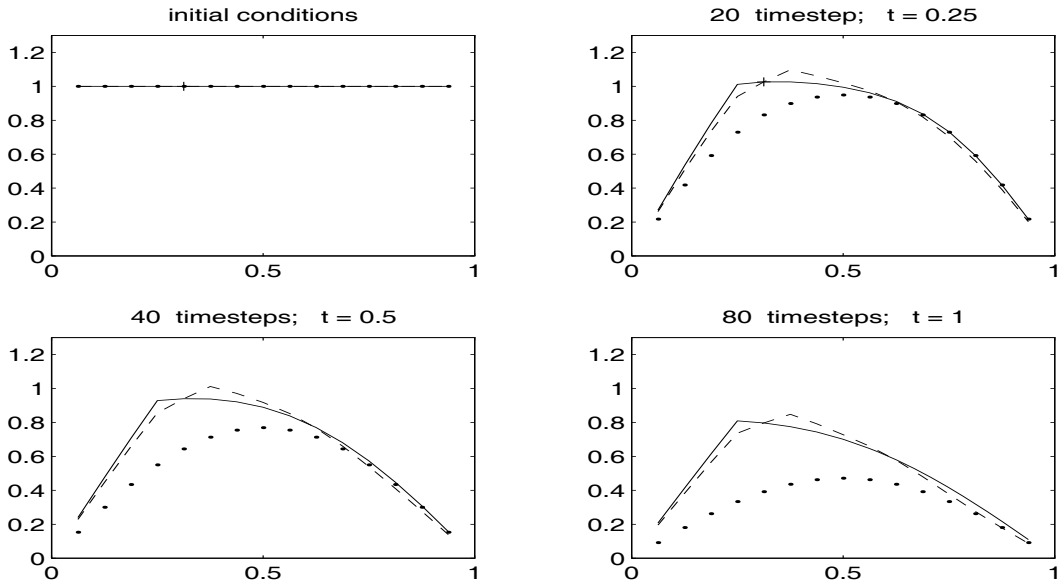

Figure 5.6: As Fig. 5.5, but using only 1 observation.

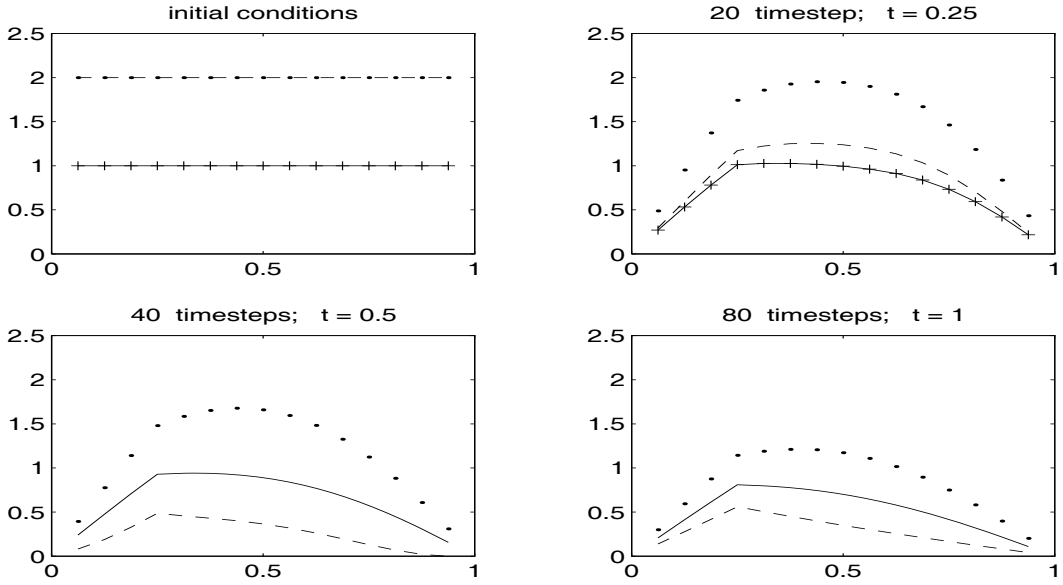

Figure 5.7: Variational assimilation using the correction term as the control vector, with $q = 0$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 15 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

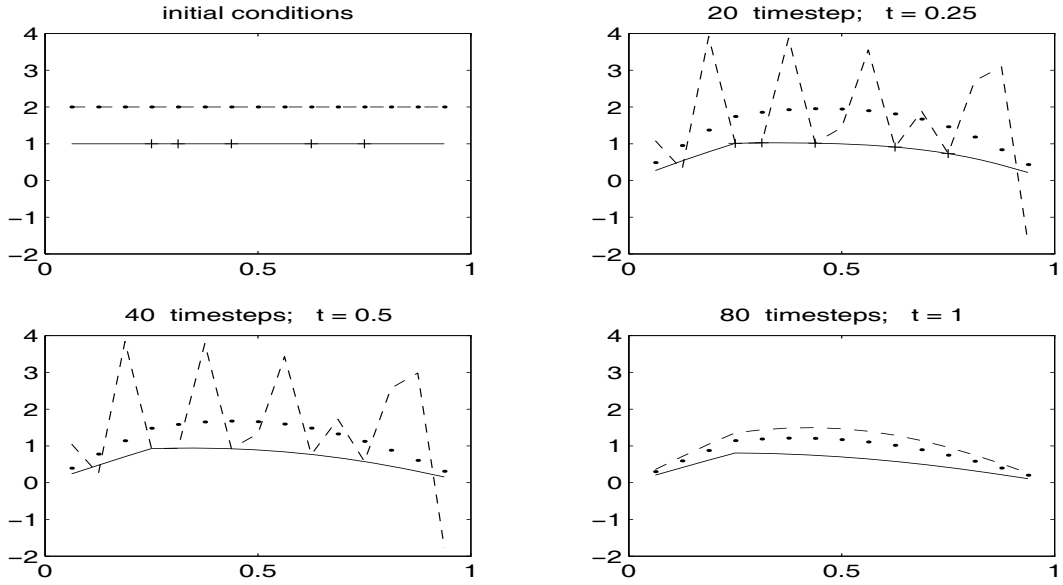

Figure 5.8: As Fig. 5.7, but using only 5 observations.

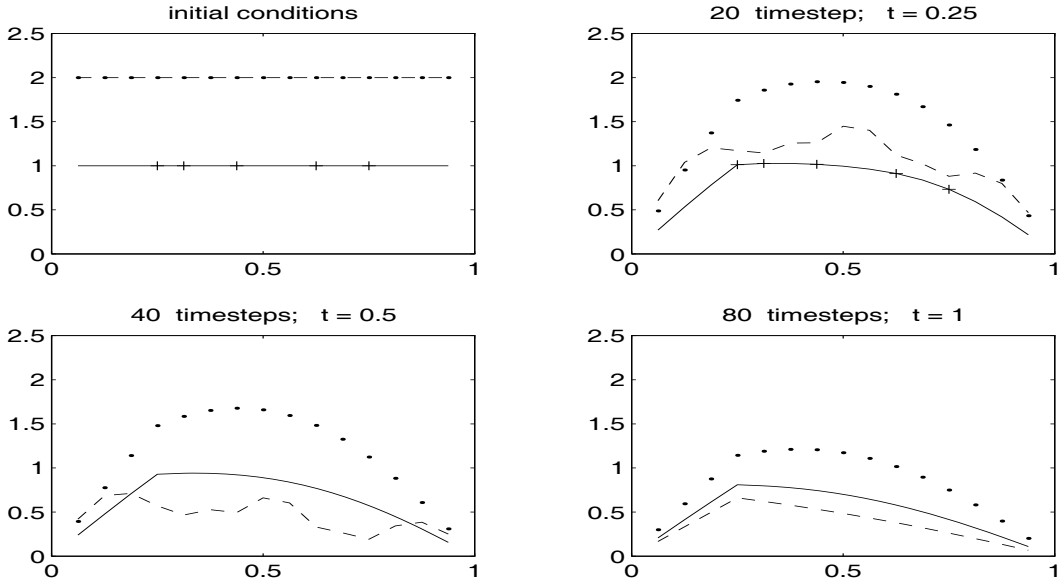

Figure 5.9: Variational assimilation using the correction term as the control vector, with $q = 1$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 5 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

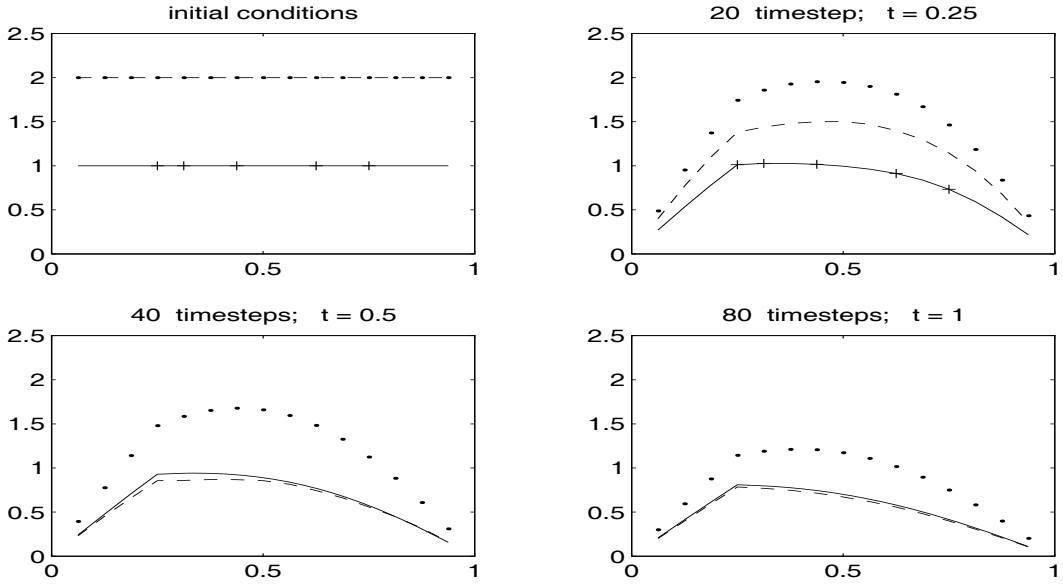

Figure 5.10: As Fig. 5.9, but using $q = 50$.

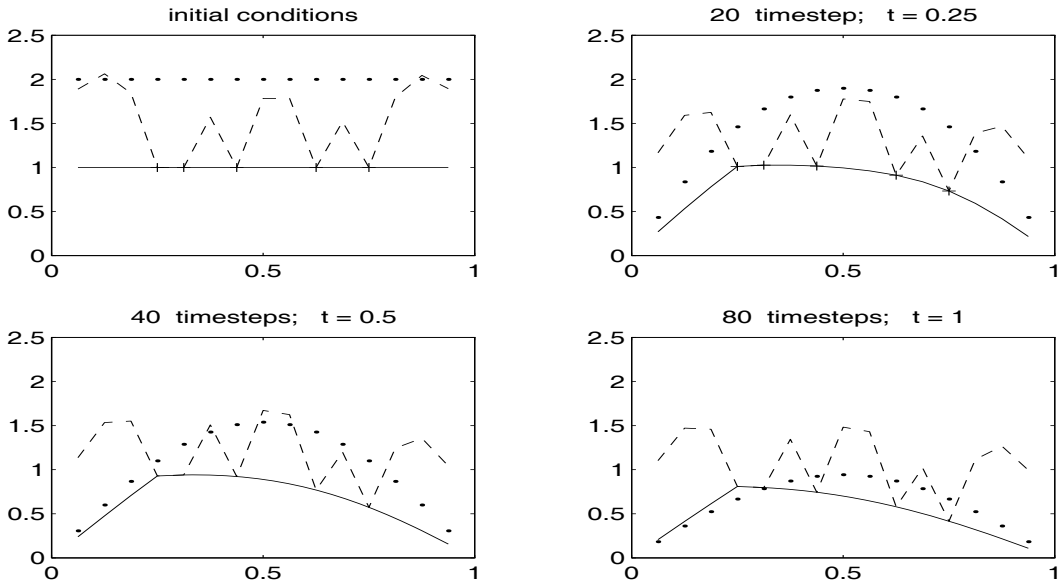

Figure 5.11: Variational assimilation using both the initial state and the correction term as the control vectors, with $q = 0$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 5 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

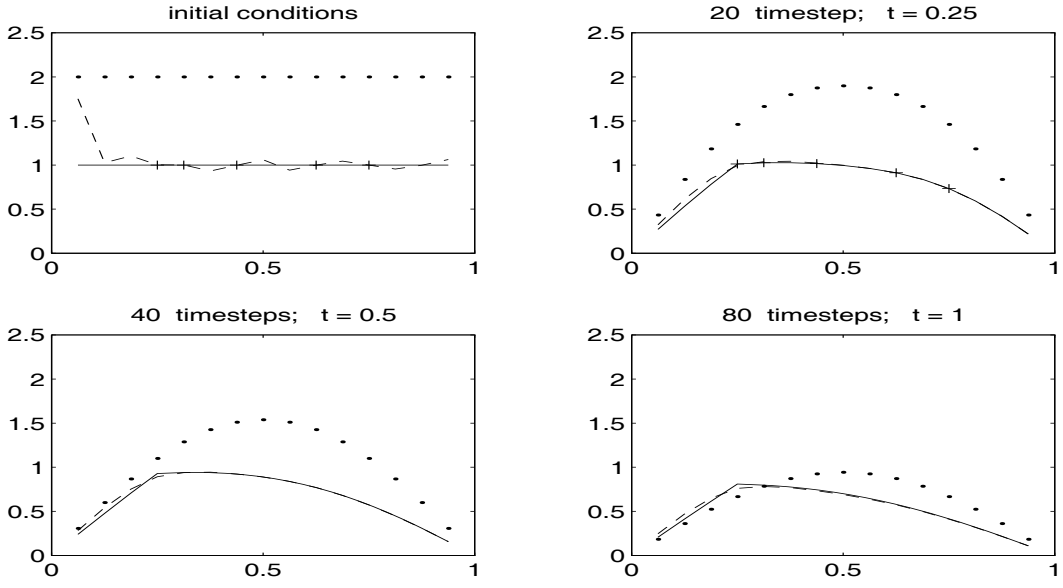

Figure 5.12: As Fig. 5.11, but using $q = 1$.

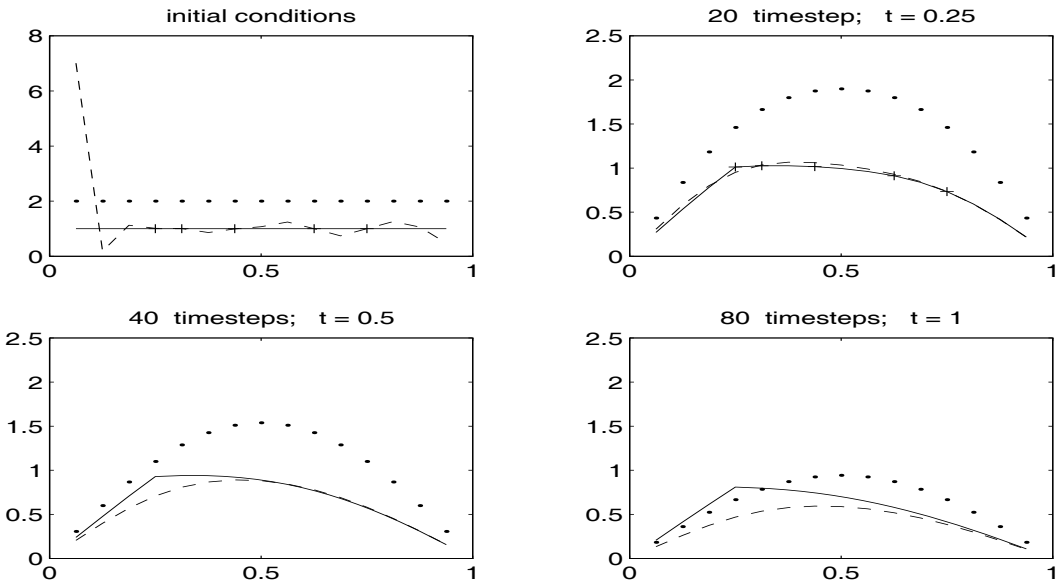

Figure 5.13: As Fig. 5.11, but using $q = 50$.

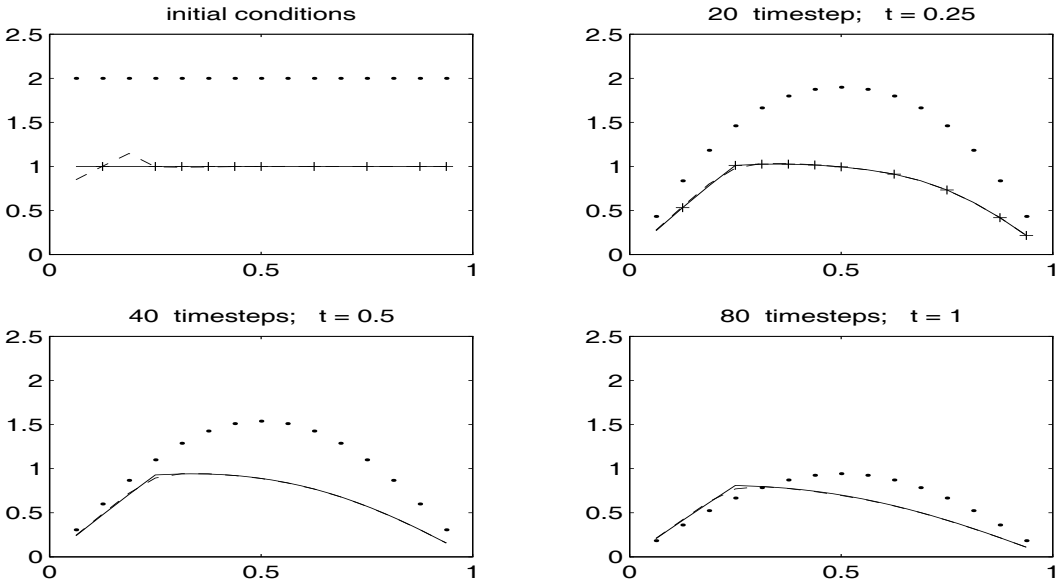

Figure 5.14: As Fig. 5.11, but using $q = 1$ and 10 observations.

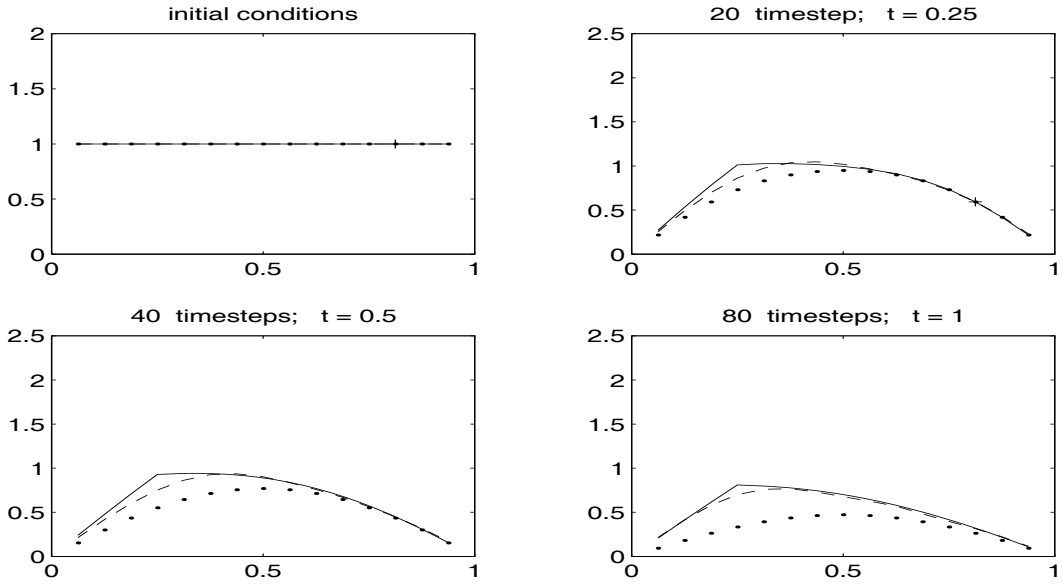

Figure 5.15: Variational assimilation using the correction term as the control vector, with $q = 0$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 1 observation, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

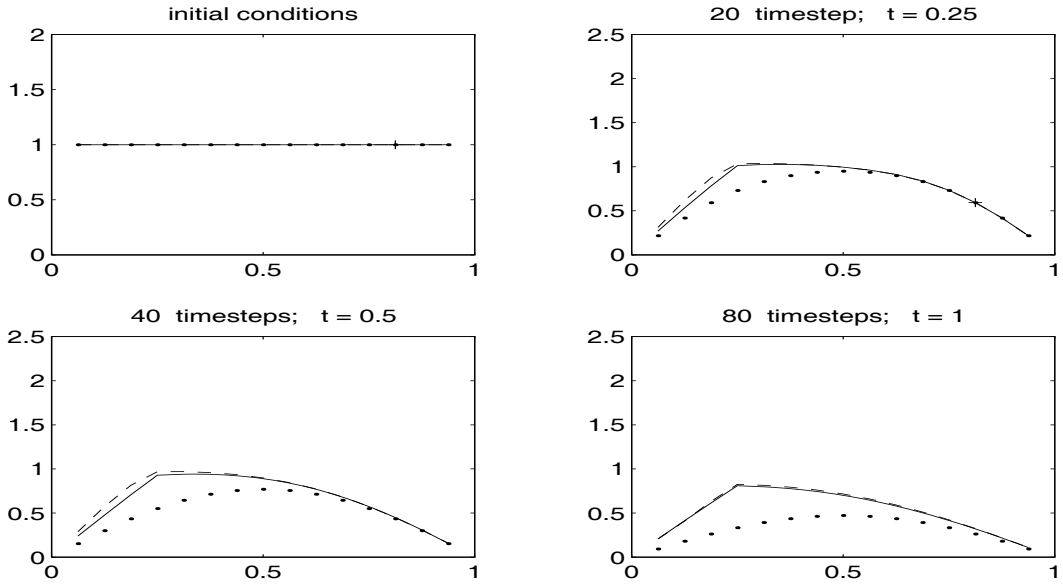

Figure 5.16: As Fig. 5.15, but with the dimension of the correction term vector reduced to $m = 5$.

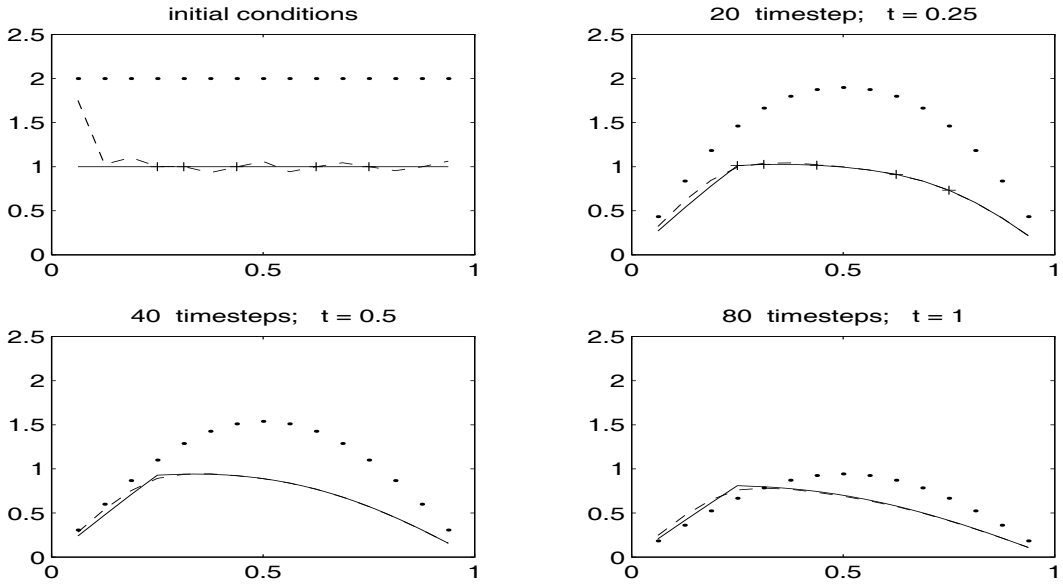

Figure 5.17: Variational assimilation using both the initial state and the correction term as the control vectors, with $q = 1$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 5 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

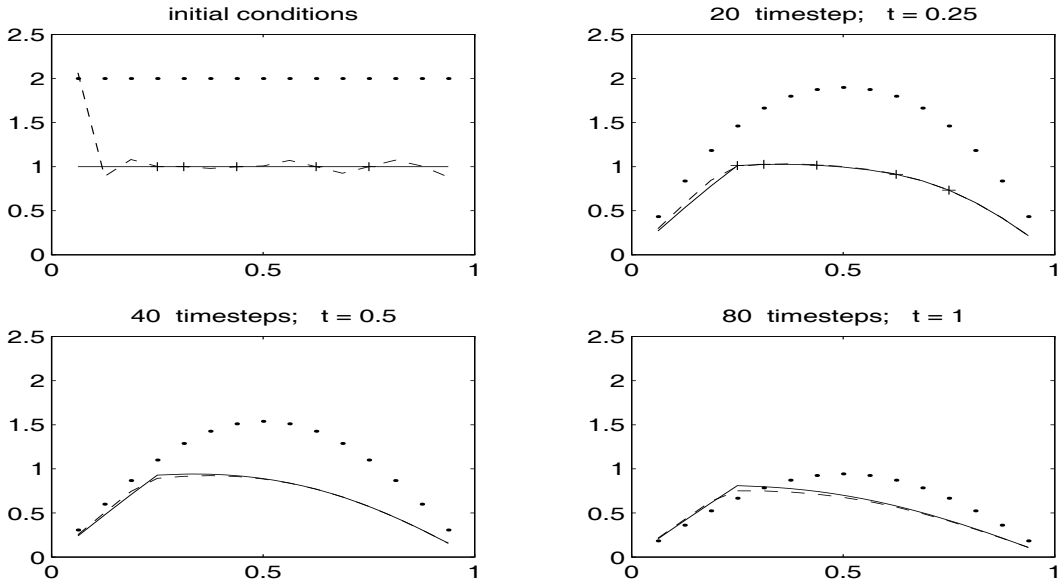

Figure 5.18: As Fig. 5.17, but with the dimension of the correction term vector reduced to $m = 5$.

5.5 Summary and conclusions

In this chapter we have looked at both theoretical and practical aspects of using the correction term as the control vector compared with using the initial state as the control vector, and also of using both control vectors together. Here we briefly summarize the theoretical results of Section 5.2, and give conclusions from the experiments described in Sections 3 and 4.

5.5.1 Summary of the theoretical results

In Section 5.2 we considered from a theoretical point of view, uniqueness of the 4D variational assimilation problem using each control vector. We were particularly interested to see when it is possible to determine each control vector from the observations alone, exactly in the case of observations with no error, or in a least squares sense in the case of imperfect observations.

The issue of uniquely determining an initial state from observations is addressed in the concept of observability. The paper by Zou et. al. [91] includes a proof that in the continuous, time invariant case, complete observability of the system is a sufficient condition for a unique solution of the 4D variational assimilation problem, Problem $\mathcal{A}_{IS}$. Here we used the concept of complete N-step observability at the initial time t_0, which is both a necessary and sufficient condition for uniqueness in the discrete, time varying case. This result is applicable to an assimilation system in which the number and type of observations vary in time. Even when applied to a time-invariant system, this result linking uniqueness to complete N-step observability tells us more than the corresponding result on complete observability if the number N of timesteps in the assimilation interval is smaller than the dimension n of the model state.

We then addressed the issue of whether it is possible to determine uniquely a constant model input (a correction term representing model error in our context) from observations on an assimilation interval. Interestingly, in this case complete N-step observability at time t_0 is neither a necessary nor a sufficient condition for uniquely determining a constant input from the observations, as we showed

using simple counter-examples. This means that in some cases it is possible to uniquely determine the correction term but not the initial state from a given set of observations, and vice versa. However, we also gave an example for the time-invariant case in which it is possible to uniquely determine the correction term from the observations if and only if it is possible to uniquely determine the initial conditions.

If both control vectors are used together, it is possible to write the original system in terms of an augmented system, in which the augmented initial state, consisting of the original initial state and the correction term, is the augmented control vector. Hence, a necessary and sufficient condition for being able to determine both the original initial state and the correction term from the observations is that the augmented system is completely N-step observable at time t_0. We looked at conditions for complete N-step observability of the augmented system in terms of complete N-step observability of the original system and showed, in particular, that for a time-invariant system, it is possible to determine the augmented control vector from the observations if and only if a full set of observations is used. In the time varying case, however, it is possible to determine the augmented control vector from the observations if there is a full set of observations at times t_0 and t_1.

In each case, adding a background estimate of the control vector to the cost function can guarantee a unique solution when the observational data cannot. This is known for data assimilation using the initial state as the control vector, but a background term was not included in published work on the correction term technique. We also showed that if both control vectors are used, then if the original system is completely N-step observable, it is only necessary to add a background estimate of the correction term.

In general, it is not possible to check complete N-step observability in the context of operational data assimilation in NWP or in oceanography, although in some cases we know whether or not this condition holds. In other applications of data assimilation, however, it is possible to check this condition. However, even when we do not know whether or not a system is completely N-step observable, these theoretical results give us insight into the comparative ability of the variational

assimilation method using different control vectors to obtain information from the observations. In particular, we showed that conditions for determining the initial state and those for determining the correction term from the observations are the same in a time-invariant system with N large enough, although they are not the same in general; that a necessary but not sufficient condition for both control vectors to be uniquely determined from the data is that each of them can be determined individually; and that in the time-invariant case it is not possible to determine both control vectors unless a full set of observations is used.

5.5.2 Conclusions from the experiments

The experiments described in this chapter show that just as it is possible to reconstruct the true solution from an unknown initial state with a perfect model and a sufficient number of perfect observations, so it is possible to reconstruct the true solution from an imperfect model with known initial state and a sufficient number of observations if model error is constant in time. In these experiments, when only one or two observations were used at each timestep, the solution was inaccurate, even though the theory shows that we should be able to obtain exact results in these cases also. This is probably due to numerical rounding error in the descent iteration procedure.

From these results, we can conclude that the correction term technique might work well to compensate for sources of model error which are approximately constant over the assimilation interval, such as forcing terms which do not depend on the model state, and misspecified, constant boundary conditions. When an imperfect model is used, using the correction term obtained in the assimilation in a subsequent forecast greatly improves the forecast, as found in earlier studies on the correction term technique.

If the initial state is the only control vector and an imperfect model is used, then the best fit to the true solution is in the middle of the assimilation interval, and there is little gain in accuracy in an ensuing forecast since this is carried out with an imperfect model. Similarly, using only the correction term as the control vector can compensate for the effect of errors in the initial conditions, although in this context

the correction term should not be included in an ensuing forecast.

In all experiments with unknown initial state in which the correction term was used as a control vector, using a background term with a large weight constraining the correction term to be small (ie, a large value of q) was vital for sensible results where fewer than the full set of observations are available, although the background term is not needed for uniqueness in all these cases. This fact was revealed because in our idealized experiments we were able to compare our results with the true solution away from the observation positions. Previous published work on the correction term technique did not include the background term.

Using a background term was also useful, although not necessary, when the correction term was being used to correct for model error, since including it speeds up the descent procedure slightly. In this case, however, using a value of q which was too large had a detrimental impact on the results. This problem of needing a small value of q to deal with the constant component of model error, and a large value of q to deal with other forms of forecast error, such as errors in the initial state, might not be such an issue in cases in which the errors dealt with are not so drastic. In these experiments we used very large model error and initial state errors for exaggerated results.

Using both control vectors together it was possible to obtain very good results in the presence of a wrong initial state, an imperfect model, or both. However, this was at the cost of significantly more descent algorithm iterations, typically 60-80 iterations with both control vectors compared with just 15-25 using just one of them, although these results were required to satisfy a very strict stopping criterion.

Reducing the dimension of the correction term control vector increased the efficiency of the method, since fewer iterations of the descent algorithm were required in this case. This is particularly significant when both control vectors are used together, since in this case the high number of iterations required was reduced by around a half. Reducing the dimension of the correction term might be appropriate if the correction term is only required to counteract the effects of model error which are known to be localized to some area.

From these experiments, two immediate issues arise which require further atten-

tion. The first is the problem of reducing the number of iterations needed when both control vectors are used. This could be achieved by suitable preconditioning of the descent process, but we do not take this any further in this work. Secondly, the question arises of how well the correction term technique would work in the presence of model error which is not constant in time, especially model error which depends on the model state. We examine this in the experiments of the next two chapters.

Chapter 6

Accounting for model error in variational assimilation

We start this chapter with a discussion on the problem of how to account for model error in data assimilation. In particular, we note that the assumptions made on model error in Kalman filtering theory have theoretical and practical limitations. We consider a more general form of model error which has serially correlated and serially uncorrelated components, and we give several different examples of how we might represent model error.

We show that the technique of *state augmentation* provides a useful tool for accounting for model error in data assimilation. Using this technique, the aim in data assimilation is to estimate serially correlated components of model error along with the model state. This leads to a generalised form of Problem $\mathcal{LS}$ of Chapter 4, in which we allow for serially correlated model error. In this context, it is possible to give a statistical interpretation of the correction term technique. The state augmentation approach also provides a way to generalize the correction term technique to represent model error that changes with the state evolution, rather than model error that is constant in time. We refer to this as the "evolving correction term technique".

We conclude this chapter with experiments using a simple model in which the model error changes with the model state. In this case, the usual correction term technique does not compensate for the effects of model error at all, but using the

evolving correction term as a control variable produces a significant improvement in the results.

The theoretical part of our work described in this chapter has been published in a shorter form in [41].

6.1 Background on representing model error

In 4D variational assimilation using the strong constraint approach, which is currently being developed for operational application in meteorological centres, model error is neglected. Recently, however, the problem of how to account for model error in variational assimilation in a cost effective way has begun to receive more attention [61].

Studies in predictability which explicitly attempt to represent the effects of model error on forecast error, [12],[22],[85] show that the impact of model error on forecast error in meteorological models is indeed significant. The study in [22] leads to the conclusion that the predictability limit of a forecast might be extended by two or three days if model error were eliminated. However, there is a lack of quantative information on model error in such forecast models, even of its size relative to that of the model state. Hence, the problem arises of how to represent model error in data assimilation.

The Kalman filter does account for model error, and in the Standard Kalman filter model error is treated as serially uncorrelated, unbiased random error. In Chapter 4, Section 4.4 we also discussed other approaches to weak constraint variational assimilation which make the same assumptions about model error. An interesting paper by Dee [25] however, questions the validity of this representation of model error. Using an analysis of model error similar to that given in Chapter 2, Subsection 2.1.2 here, he argues that since model error in general depends on the model state, it is likely to be serially correlated. In Chapter 3, Section 3.2 we gave background on how the Kalman filter can be modified to deal with serially correlated model error, but this involves a large increase in the expense of the method, which (unless in simplified form) is already thought to be too expensive for operational

application in data assimilation.

Another major problem with using a stochastic representation of model error in data assimilation is that it is very difficult to model the error covariance matrix, since the statistics of model error are largely unknown. Dee [25] argues that it is this huge information requirement, rather than the large computational cost, that is the real obstacle to successful implementation of the Kalman filter. He further argues that it doesn't make sense to expend huge amounts of effort in propagating the error covariance matrices when the statistical assumptions made on model error are suspect. He concludes that until further information about the statistics of model error are available, the advantages of the Kalman filter over other data assimilation schemes are "strictly hypothetical".

Here, we are concerned with how to account for model error in variational assimilation. The weak constraint approach to variational assimilation does allow for model error, and in Chapter 4, Section 4.4 we reviewed some of the methods for solving this minimization problem which have been proposed for data assimilation. Generally, these methods make the same statistical assumptions on model error as made in the Kalman filter, and so the theoretical problems raised above apply here too.

The correction term technique provides a way of allowing to some extent for model error in variational assimilation. In Chapter 5 we saw that this works very well for model error that is constant in time, but this is of course not generally the case. Papers on the use of the correction term technique refer to the correction term as representing "model bias" or as representing "average" model error, but published work has not provided a theoretical statistical interpretation of the analysis the correction term technique provides, as has been done in the case of the strong constraint approach (ie, that under certain statistical assumptions it represents the "most likely solution", if the model can be assumed to be perfect).

In Section 6.2 of this chapter, we consider a general representation of model error that can be used to represent each of the forms of model error that have been suggested for use in data assimilation, and which could also represent other forms. We consider how the technique of *state augmentation* can be used to estimate

a serially correlated component of model error along with the model state. In Section 6.3 we give a generalised version of Problem $\mathcal{LS}$ (the general least squares problem for variational assimilation introduced in Chapter 4) which allows for the state augmentation approach. In this context we can interpret the correction term technique in a statistical way.

6.2 State augmentation

We consider the nonlinear model

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \boldsymbol{\varepsilon}_k, \qquad k = 0, .., N-1 \tag{6.1}$$

as defined in (2.4), where $\mathbf{x}_k \in \mathbb{R}^n$ and $\boldsymbol{\varepsilon}_k \in \mathbb{R}^n$ are the model state and the model error at time t_k. In Chapter 5 on the correction term technique, we considered an approximation of the model error term $\boldsymbol{\varepsilon}_k$ of the form

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}, \tag{6.2}$$

where the $B_k \in \mathbb{R}^{n \times m}$ are prescribed matrices, and $\mathbf{e} \in \mathbb{R}^m$ is a constant correction term to be determined. In the correction term technique, $\mathbf{e}$ is used as a control vector in the minimization. If both the initial state $\mathbf{x}_0$ and the correction term $\mathbf{e}$ are used as control vectors, we saw that it is convenient for theoretical purposes to write the system as an equivalent augmented system. This technique of *state augmentation* is sometimes used in the control theory literature as a way of estimating, along with the model state, unknown, constant model parameters, as discussed in the text by Jazwinski [44] Chapter 8, Section 4, or of estimating a serially correlated component of random model error, as discussed in the text by Gelb [31] Chapter 3, Section 8. As we will show in this chapter, this technique therefore provides a convenient way to account for serially correlated model error in data assimilation.

6.2.1 A general formulation of model error

We now consider a more general form of model error than (6.2), which also has a random component. Following [31], we consider a stochastic form of model error,

which is made up of serially correlated errors and serially uncorrelated random errors. We therefore write

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.3}$$

where the vectors $\mathbf{q}'_k$ are serially uncorrelated, random n-vectors, the matrices $B_k \in \mathbb{R}^{n\times m}$ are prescribed matrices as before, and the vectors $\mathbf{e}_k \in \mathbb{R}^m$ represent the *serially correlated component of model error*. We suppose that we know how the error $\mathbf{e}_k$ evolves in time, and for now write this in a very general form,

$$\mathbf{e}_{k+1} = \mathbf{g}_k(\mathbf{x}_k, \mathbf{e}_k) + \mathbf{q}''_k, \tag{6.4}$$

where $\mathbf{g}_k : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ is some function to be specified, and the vectors $\mathbf{q}''_k$ are serially uncorrelated random m-vectors.

As we discussed in Section 6.1, we know very little about the form of the model error, and in practice will have to specify (6.3), (6.4) in a very simple form which reflects any knowledge of model error we do have. We give a few such examples in Subsections 6.2.2 and 6.2.3. Using this general formalism, however, we can allow for model error which depends on the model state, and for other types of model error discussed earlier.

The model system can now be written

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.5}$$

$$\mathbf{e}_{k+1} = \mathbf{g}_k(\mathbf{x}_k, \mathbf{e}_k) + \mathbf{q}''_k, \qquad k = 0, .., N-1 \tag{6.6}$$

or as the equivalent *augmented system*

$$\mathbf{w}_{k+1} = \tilde{\mathbf{f}}_k(\mathbf{w}_k) + \mathbf{q}_k, \qquad k = 0, .., N-1 \tag{6.7}$$

in which $\mathbf{w}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{e}_k \end{pmatrix} \in \mathbb{R}^{(n+m)}$ is the *augmented state vector*, $\tilde{\mathbf{f}}_k : \mathbb{R}^{(n+m)} \to \mathbb{R}^{(n+m)}$ is a nonlinear function, and $\mathbf{q}_k$ is a random $(n+m)$-vector. The aim of the data assimilation problem for the augmented system is to estimate the augmented state $\mathbf{w}_k$. Before discussing how to do this, we first give some examples of how we may represent the model error term.

6.2.2 Examples of how model error can be specified

i) Serially uncorrelated model error

Setting all the $\mathbf{e}_k$ in (6.3) to zero we have

$$\boldsymbol{\varepsilon}_k = \mathbf{q}'_k, \tag{6.8}$$

in which case model error is a serially uncorrelated random vector, as assumed in the Standard Kalman filter.

ii) Constant model bias error

Setting

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.9}$$

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \tag{6.10}$$

allows for a constant vector of unknown "dynamical parameters" as discussed in [44]. If this form of model error is purely deterministic (ie, $\mathbf{q}'_k = 0$), this represents the correction term technique of Chapter 5. In Derber's paper [26] introducing the correction term technique, the matrices B_k were the $n \times n$ identity multiplied by a time-varying scalar and by the time-step length Δt, to reflect the rôle of this form of model error as a correction to the time derivative of the model equations. As discussed in Chapter 5, we expect this form of model error to be appropriate for representing constant errors in the forcing or in the boundary conditions.

iii) Model error evolving with model evolution

In Section 6.1, we discussed that model error is likely in general to depend on the true model state, and hence to change with the flow. In this case model error evolution might be approximated by

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.11}$$

$$\mathbf{e}_{k+1} = G_k \mathbf{e}_k, \tag{6.12}$$

where $G_k \in \mathbb{R}^{m \times m}$ represents a simplified form of the model state evolution. This might be an appropriate approximation to model error evolution if model error

represents truncation error. This is similar to the form of serially correlated model error which was suggested in the paper by Daley [23] in formulating a Kalman filter allowing for serially correlated model error, which we discussed in Chapter 3, Section 3.2.

The matrices B_k in (6.3) allow for a serially correlated component of model error with dimension m which may be less than or greater than the dimension n of the model state. In Chapter 5 we showed how using $m < n$ can lead to greater efficiency in the correction term technique if the source of model error is known to be localized. We now consider how including the possibility that $m > n$ can allow for greater flexibility in the specification of model error.

We may partition the serially correlated component of model error in r subvectors of dimension s (where $rs = m$), and write

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.13}$$

$$\mathbf{e}_{k+1} = \mathbf{g}_k(\mathbf{w}_k) + \mathbf{q}''_k, \tag{6.14}$$

where this time

$$B_k = (B_k^{(1)}, B_k^{(2)}, .., B_k^{(r)}), \qquad \mathbf{e}_k = \begin{pmatrix} \mathbf{e}_k^{(1)} \\ \vdots \\ \mathbf{e}_k^{(r)} \end{pmatrix}, \tag{6.15}$$

where $\mathbf{e}_k^{(1)}, .., \mathbf{e}_k^{(r)} \in \mathbb{R}^s$ and $B_k^{(1)}, .., B_k^{(r)} \in \mathbb{R}^{n \times s}$. The following examples illustrate how this generalization might be useful.

iv) Model error growing in time

Here, rather than using a constant correction term to represent model error as in the correction term technique, we allow for a correction term which can increase or decrease linearly in time. In this case model error has the form

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.16}$$

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \tag{6.17}$$

with

$$B_k = (B_k^{(1)}, \Delta t B_k^{(2)}), \qquad \mathbf{e}_k = \begin{pmatrix} \mathbf{e}_k^{(1)} \\ \mathbf{e}_k^{(2)} \end{pmatrix}. \tag{6.18}$$

This form of model error is referred to in [31] as a "random ramp", since its initial size and rate of change are to be determined.

v) Combination of Examples i) and ii)

We suppose the model error has the form

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.19}$$

$$\mathbf{e}_{k+1} = \tilde{G}_k \mathbf{e}_k, \tag{6.20}$$

with

$$B_k = (B_k^{(1)}, B_k^{(2)}), \qquad \mathbf{e}_k = \begin{pmatrix} \mathbf{e}_k^{(1)} \\ \mathbf{e}_k^{(2)} \end{pmatrix}, \quad \tilde{G}_k = \begin{pmatrix} G_k & 0 \\ 0 & I \end{pmatrix}, \tag{6.21}$$

where G_k is as defined in (6.12). In this we can allow for model error with a constant component and a component which changes with model evolution.

vi) Spectral form of model error

In this case we suppose that model error has the form

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.22}$$

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \tag{6.23}$$

with

$$B_k = (B_k^{(1)}, B_k^{(2)} \sin\left(\frac{k}{N\tau}\right), B_k^{(3)} \cos\left(\frac{k}{N\tau}\right)), \qquad \mathbf{e}_k = \begin{pmatrix} \mathbf{e}_k^{(1)} \\ \mathbf{e}_k^{(2)} \\ \mathbf{e}_k^{(3)} \end{pmatrix}, \tag{6.24}$$

where τ is a constant which might be chosen bearing in mind the timescale on which model error is expected to vary, for example a diurnal timescale.

vii) Piecewise constant model error

Here we suppose that the assimilation interval $[t_0, t_N]$ is broken into r subintervals over which model error is represented by different constant correction terms. For convenience, we suppose here that N is a multiple of r, so that we can represent model error as

$$\boldsymbol{\varepsilon}_k = B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.25}$$

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \tag{6.26}$$

with

$$B_k = (s_k^{(1)} B_k^{(1)}, s_k^{(2)} B_k^{(2)}, .., s_k^{(r)} B_k^{(r)}), \qquad \mathbf{e}_k = \begin{pmatrix} \mathbf{e}_k^{(1)} \\ \vdots \\ \mathbf{e}_k^{(r)} \end{pmatrix}, \tag{6.27}$$

where the scalars $s_k^{(i)}$ are given by

$$s_k^{(i)} = \begin{cases} 1 & \text{for} \quad k = \frac{(i-1)N}{r}, .., \frac{iN}{r} - 1 \\ 0 & \text{otherwise.} \end{cases} \tag{6.28}$$

If $r = N$, $B_k = I$ and $\mathbf{q}'_k = 0$ then we estimate N serially uncorrelated model error terms.

6.2.3 Problem $\mathcal{LS}$ for the augmented system

Using our most general form of model error, the nonlinear system is

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + B_k \mathbf{e}_k + \mathbf{q}'_k, \tag{6.29}$$

$$\mathbf{e}_{k+1} = \mathbf{g}_k(\mathbf{w}_k) + \mathbf{q}''_k \qquad k = 0, .., N-1, \tag{6.30}$$

which can equivalently be written in as the augmented system

$$\mathbf{w}_{k+1} = \tilde{\mathbf{f}}_k(\mathbf{w}_k) + \mathbf{q}_k, \qquad k = 0, .., N-1, \tag{6.31}$$

where $\mathbf{w}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{e}_k \end{pmatrix} \in \mathbb{R}^{n+m}$ is the augmented state vector, $\mathbf{q}_k = \begin{pmatrix} \mathbf{q}'_k \\ \mathbf{q}''_k \end{pmatrix}$ is a random $(n+m)$-vector, and $\tilde{\mathbf{f}}_k : \mathbb{R}^{n+m} \to \mathbb{R}^{n+m}$ is a nonlinear function describing

the evolution of the augmented state vector. We suppose that as before, we have observations given by

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \boldsymbol{\delta}_k, \qquad k = 0, .., N-1 \tag{6.32}$$

as defined in (2.6). We define $\tilde{\mathbf{h}}_k : \mathbb{R}^{n+m} \to \mathbb{R}^{p_k}$ by $\tilde{\mathbf{h}}_k(\mathbf{w}_k) = \mathbf{h}_k(\mathbf{x}_k)$. We assume that the quantities $\mathbf{w}_0$, $\mathbf{q}_k$ and $\boldsymbol{\delta}_k$, $k = 0, .., N-1$, are not correlated with each other. We suppose that $\mathbf{q}_k$ has a positive definite covariance matrix $S_k \in \mathbb{R}^{m \times m}$, and that we have a prior estimate or "background" estimate $\mathbf{w}_0^b$ of $\mathbf{w}_0$, and that the covariance matrix of the errors $(\mathbf{w}_0 - \mathbf{w}_0^b)$ is given by the nonsingular matrix $\tilde{P}_0 \in \mathbb{R}^{(n+m)\times(n+m)}$. As before, we suppose that the covariance matrix of the observational error $\boldsymbol{\delta}_k$ is given by the positive definite matrix $R_k \in \mathbb{R}^{p_k \times p_k}$.

For this augmented system, with observations (6.32) and prior estimate $\mathbf{w}_0^b$, the general least squares problem for estimating the augmented state $\mathbf{w}_0, ..., \mathbf{w}_N$ is

Problem $\mathcal{LSA}$

Minimize, with respect to $\mathbf{w}_0, .., \mathbf{w}_N, \mathbf{q}_0, .., \mathbf{q}_{N-1}$

$$\begin{aligned} \mathcal{J} &= \frac{1}{2}(\mathbf{w}_0 - \mathbf{w}_0^b)^T \tilde{P}_0^{-1}(\mathbf{w}_0 - \mathbf{w}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\tilde{\mathbf{h}}_j(\mathbf{w}_j) - \mathbf{y}_j)^T R_j^{-1}(\tilde{\mathbf{h}}_j(\mathbf{w}_j) - \mathbf{y}_j) \\ &\quad + \frac{1}{2}\sum_{j=0}^{N-1} \mathbf{q}_j^T S_j^{-1} \mathbf{q}_j, \end{aligned} \tag{6.33}$$

subject to (6.31).

Problem $\mathcal{LSA}$ is a generalization of Problem $\mathcal{LS}$ which allows for a more general form of model error. Hence, if the errors $\{\mathbf{q}_k\}$ and $\{\boldsymbol{\delta}_k\}$ are Gaussian and unbiased, and if the system (6.31), (6.32) is linear, then the solution of Problem $\mathcal{LSA}$ represents the "most likely solution" as defined in Chapter 4, Section 4.4. The methods outlined in Chapter 4 for solving Problem $\mathcal{LS}$ could also be applied to this generalised version. If the Kalman filter is used to solve Problem $\mathcal{LSA}$ for a linear system, then if model error is assumed to have the form given in Example iii) of Section 6.2.2, we have the method of accounting for serially correlated model error in Kalman filtering outlined in Chapter 3, Section 3.2. This approach, however, involves propagating extra covariance matrices, and so is much more expensive than

the standard Kalman filter. It is possible to allow for serially correlated model error in the Kalman filter without actually estimating the serially correlated component of the model error $\mathbf{e}_k$, [44], [23]. In this case the state dimension is still only n, but we are not able to improve an initial estimate of the $\mathbf{e}_k$ during the assimilation, as we do using the state augmentation approach. The representer method seems a promising way of accounting for model error in 4D data assimilation at a reasonable cost. If applied to Problem $\mathcal{LSA}$ rather than Problem $\mathcal{LS}$, however, this method can also allow for serially correlated model error.

If we neglect the serially uncorrelated part of the model error, $\mathbf{q}_k$, then we can use the augmented initial state $\mathbf{w}_0$ as the control vector, as we discuss next.

6.3 A generalized correction term technique

In the strong constraint approach to 4D variational assimilation outlined in Chapter 4, in which the initial state is used as the control vector, model error is neglected. Since model error is not negligible in reality, this method finds only an approximation to the optimal solution. If we attempt to estimate an augmented state which includes serially correlated components of model error, however, it should be possible to obtain a better approximation to the optimal solution. We therefore attempt to solve Problem $\mathcal{LSA}$ neglecting the random errors $\mathbf{q}_k$. The accuracy of the solutions we obtain will depend on how well the serially correlated component of model error we estimate represents the actual model error.

In this case, we estimate the model state $\mathbf{x}_k$ and a correction term $\mathbf{e}_k$ representing serially correlated model error, using $\mathbf{w}_0 = \begin{pmatrix} \mathbf{x}_0 \\ \\ \mathbf{e}_0 \end{pmatrix}$ as a control vector. This can be seen as a generalization of the correction term technique in which the correction term $\mathbf{e}_k$ may evolve in time.

Hence, the correction term technique provides an optimal solution to Problem $\mathcal{LSA}$ assuming that model error is represented by a constant bias and does not have a serially uncorrelated component. Viewing the correction term technique as a method of solving Problem $\mathcal{LSA}$ again points out the theoretical importance

of including a background term for $\mathbf{e}_0$ in the cost function.

Adjoint equations and gradients

We now follow the development of Chapter 4, Section 4.1 to solve Problem $\mathcal{LSA}$ with $\mathbf{q}_k = 0$ using the augmented initial state $\mathbf{w}_0$ as the augmented control vector. In this case, the Lagrangian is given by

$$\begin{aligned}\mathcal{L} &= \frac{1}{2}(\mathbf{w}_0 - \mathbf{w}_0^b)^T \tilde{P}_0^{-1}(\mathbf{w}_0 - \mathbf{w}_0^b) + \frac{1}{2}\sum_{j=0}^{N-1}(\tilde{\mathbf{h}}_j(\mathbf{w}_j) - \mathbf{y}_j)^T R_j^{-1}(\tilde{\mathbf{h}}_j(\mathbf{w}_j) - \mathbf{y}_j) \\ &\quad + \sum_{j=0}^{N-1} \boldsymbol{\nu}_{j+1}^T(\mathbf{w}_{j+1} - \tilde{\mathbf{f}}_j(\mathbf{w}_j)), \end{aligned} \tag{6.34}$$

where the $\boldsymbol{\nu}_k \in \mathbb{R}^{(n+m)}$ are vectors of Lagrange multipliers.

The adjoint equations are given by

$$\boldsymbol{\nu}_k = \tilde{F}_k^T(\mathbf{w}_k)\boldsymbol{\nu}_{k+1} - \tilde{H}_k^T(\mathbf{w}_k)R_k^{-1}(\tilde{\mathbf{h}}_k(\mathbf{w}_k) - \mathbf{y}_k), \qquad k = N-1, .., 1 \tag{6.35}$$

with

$$\boldsymbol{\nu}_N = 0, \tag{6.36}$$

where $\tilde{F}_k \in \mathbb{R}^{(n+m)\times(n+m)}$ and $\tilde{H}_k \in \mathbb{R}^{p_k\times(n+m)}$ are the Jacobians of $\tilde{\mathbf{f}}_k$ and $\tilde{\mathbf{h}}_k$ with respect to $\mathbf{w}_k$.

The gradient of $\mathcal{L}$ with respect to the augmented control vector $\mathbf{w}_0$ is

$$\nabla_{\mathbf{w}_0}\mathcal{L} = \tilde{P}_0^{-1}(\mathbf{w}_0 - \mathbf{w}_0^b) - \boldsymbol{\nu}_0, \tag{6.37}$$

where $\boldsymbol{\nu}_0$ is defined by (6.35) with $k = 0$. Algorithm IS of Chapter 4, Section 4.1 may now be applied to this case.

In terms of the original system and model error equations (6.29),(6.30), the augmented equations can be written

$$\boldsymbol{\lambda}_k = F_k^T(\mathbf{x}_k)\boldsymbol{\lambda}_{k+1} + G_k^T(\mathbf{w}_k)\boldsymbol{\mu}_{k+1} - H_k^T R_k^{-1}(\mathbf{h}_k(\mathbf{x}_k) - \mathbf{y}_k) \tag{6.38}$$

$$\boldsymbol{\mu}_k = B_k^T\boldsymbol{\lambda}_{k+1} + \Gamma_k^T(\mathbf{w}_k)\boldsymbol{\mu}_{k+1} \qquad k = 0, .., N-1 \tag{6.39}$$

with

$$\boldsymbol{\lambda}_N = 0, \tag{6.40}$$

$$\boldsymbol{\mu}_N = 0, \tag{6.41}$$

where $\boldsymbol{\lambda}_k \in \mathbb{R}^n$, $\boldsymbol{\mu}_k \in \mathbb{R}^m$, and where $G_k \in \mathbb{R}^{m\times n}$ is the Jacobian of $\mathbf{g}_k$ with respect to $\mathbf{x}_k$, and $\Gamma_k \in \mathbb{R}^{m\times m}$ is the Jacobian of $\mathbf{g}_k$ with respect to $\mathbf{e}_k$. With

$$\tilde{P}_0^{-1} = \begin{pmatrix} P_0^{-1} & 0 \\ 0 & Q_0^{-1} \end{pmatrix}, \tag{6.42}$$

where $P_0 \in \mathbb{R}^{n\times n}$ and $Q_0 \in \mathbb{R}^{m\times m}$ are the covariance matrices of $(\mathbf{x}_0 - \mathbf{x}_0^b)$ and $(\mathbf{e}_0 - \mathbf{e}_0^b)$ respectively, equation (6.37) becomes

$$\nabla_{\mathbf{x}_0}\mathcal{L} = P_0^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b) - \boldsymbol{\lambda}_0, \tag{6.43}$$

$$\nabla_{\mathbf{e}_0}\mathcal{L} = Q_0^{-1}(\mathbf{e}_0 - \mathbf{e}_0^b) - \boldsymbol{\mu}_0, \tag{6.44}$$

with $\boldsymbol{\lambda}_0$ and $\boldsymbol{\mu}_0$ defined by (6.38) and (6.39) with $k = 0$.

Reduced work with a non-evolving correction term

We now note that if (6.30) has the trivial form

$$\mathbf{e}_{k+1} = \mathbf{e}_k, \tag{6.45}$$

then (6.39) may be rewritten

$$\boldsymbol{\mu}_{N-k} = \sum_{m=0}^{k-1} B_{N-m-1}^T \boldsymbol{\lambda}_{N-m} + \boldsymbol{\mu}_N, \tag{6.46}$$

and hence

$$\boldsymbol{\mu}_0 = \sum_{j=1}^{N-1} B_{j-1}^T \boldsymbol{\lambda}_j, \tag{6.47}$$

so

$$\nabla_{\mathbf{e}_0}\mathcal{L} = Q_0^{-1}(\mathbf{e}_0 - \mathbf{e}_0^b) - \sum_{j=1}^{N-1} B_{j-1}^T \boldsymbol{\lambda}_j. \tag{6.48}$$

Hence, because the model error evolution has a trivial form here, equations (6.45) and (6.39) can be eliminated, and the gradients of $\mathcal{L}$ with respect to a guess of each control vector found from a run of the original model and adjoint equations only.

Hence, there is very little extra computational effort in the procedure for calculating the gradient of $\mathcal{L}$ with respect to $\mathbf{e}_0$ along with the gradient with respect to $\mathbf{x}_0$. This is an important point, since the model run and adjoint run represent the most expensive part of the descent iteration process. More storage is needed for the

extra control vector $\mathbf{e}_0$ and its gradient, although the dimension m of these might be much less than n. The increased dimension of the augmented control vector also means that the part of the descent algorithm which uses the gradient information to improve a guess of the control vector will be more expensive. A larger problem, however, is that the conditioning of the problem using the augmented control vector approach will be altered, and as a result, more iterations and hence more model and adjoint runs may be needed, as we found in the experiments of Chapter 5 using both control vectors.

6.4 Using an evolving correction term

6.4.1 Introduction

In the experiments of Chapter 5, we saw that the correction term technique is successful in correcting for model error which behaves like a constant forcing term. Here, we consider the upwind discretization of the linear advection equation in which model error is present due to dissipation. The model error can be expressed as truncation error, and since this depends on the true model state, it will change in time with the model state. In this section, we consider the generalized correction term technique supposing that the correction term representing model error evolves with the model equations.

The model has the form

$$\mathbf{x}_{k+1} = A\mathbf{x}_k, \tag{6.49}$$

and we try to compensate for model error using an *evolving correction term* $\mathbf{e}_k \in \mathbb{R}^n$, where

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{e}_k, \tag{6.50}$$

$$\mathbf{e}_{k+1} = A\mathbf{e}_k. \tag{6.51}$$

We consider using the initial state $\mathbf{x}_0$, the initial correction $\mathbf{e}_0$ and both together as control vectors. We note that when the initial correction $\mathbf{e}_0$ is used as a control vector, the dimension of the augmented model system and its adjoint is twice that of the original system and its adjoint. We noted in the previous section that this can

be avoided if the correction term is constant. We must assess, therefore, whether the benefits in correcting for model error using the evolving correction term are worth the extra effort.

6.4.2 Description of the experiments

We first introduce the linear advection model and its discretization using the upwind scheme. We then specify what observational data is available, and the minimization algorithm used, and then state the experiments that are carried out.

The Upwind Scheme for the Linear Advection Equation

The linear advection equation on $z \in [0,1]$, $t \in [0,1]$, with periodic boundary conditions is given by

$$\frac{\partial v}{\partial t} + c\frac{\partial v}{\partial z} = 0, \tag{6.52}$$

with

$$v(0,t) = v(1,t). \tag{6.53}$$

We suppose we have initial conditions given by

$$v(z,0) = \alpha(z) = \begin{cases} -0.5 & z < 0.25, \\ 0.5 & 0.25 < z < 0.5, \\ -0.5 & z > 0.5. \end{cases} \tag{6.54}$$

The upwind scheme for the linear advection equation (6.52) with (6.53) for $c > 0$ is

$$x_j^{k+1} - x_j^k = -c\frac{\Delta t}{\Delta z}(x_j^k - x_{j-1}^k), \qquad j = 1,..J, \quad k = 0,1,..N, \tag{6.55}$$

with $\Delta z = \frac{1}{J}$, $\Delta t = \frac{1}{N}$ and $x_j^k \approx v(j\Delta z, k\Delta t)$, or

$$x_j^{k+1} = (1-\mu)x_j^k + \mu x_{j-1}^k, \tag{6.56}$$

where $\mu = c\frac{\Delta t}{\Delta z}$, with x_0^k defined to be x_J^k, and with

$$x_j^0 = \alpha(j\Delta z). \tag{6.57}$$

The scheme can be written as a matrix system as follows,

$$\mathbf{x}_{k+1} = A\mathbf{x}_k, \tag{6.58}$$

in which $\mathbf{x}_k \in \mathbb{R}^n$ is the state at time t_k, where $n = J$, and $A \in \mathbb{R}^{n\times n}$ is given by

$$A = \begin{pmatrix} (1-\mu) & 0 & & \mu \\ \mu & (1-\mu) & 0 & \\ \ddots & \ddots & \ddots & \\ 0 & & \mu & (1-\mu) \end{pmatrix}. \tag{6.59}$$

The upwind scheme is first order accurate and stable provided $\mu \leq 1$.

We run the model (6.58) with $c = 1$, using $N = 80$, $J = 40$, so $\Delta t = \frac{1}{80}$ and $\Delta z = \frac{1}{40}$. Hence, $\mu = \frac{1}{2}$ and the model state has dimension $n = 40$. Since $c = 1$, the square wave represented by the initial conditions is advected all the way round the model domain to its starting position on the time interval $[0, 1]$.

The true model state

With $\mu = 1$, the upwind discretization yields the true solution of the pde (6.52) on the model grid, ie, there is no model error. So, to compute the true model state $\mathbf{x}_k^t$ on the model grid specified above, we used the model (6.58) with $\mu = 1$, choosing $\Delta t = \frac{1}{80}$, $\Delta z = \frac{1}{80}$, and with initial conditions (6.54).

Observations

We suppose that we have error free observations at p of the 40 grid points at every timestep on the interval $[0, \frac{1}{2}]$, ie for $\frac{N}{2} = 40$ timesteps, and that after this no further observations are available. Hence, the observations are given by

$$\mathbf{y}_k = C\mathbf{x}_k^t, \qquad k = 0, ..., \frac{N}{2} - 1, \tag{6.60}$$

where the observational matrix $C \in \mathbb{R}^{p\times n}$ has a simple form since the observation positions coincide with the grid points. The positions of the observations used in each case are shown in the figures.

The minimization algorithm

The minimization algorithm used is the conjugate gradient descent method, implemented as described in Chapter 5, Section 5.3.

The experiments

We minimize the cost functional

$$\mathcal{J} = \frac{1}{2}\mathbf{e}_0^T Q_0^{-1}\mathbf{e}_0 + \frac{1}{2}\sum_{j=0}^{\frac{N}{2}-1}(C\mathbf{x}_j - \mathbf{y}_j)^T R^{-1}(C\mathbf{x}_j - \mathbf{y}_j), \tag{6.61}$$

subject to

$$\mathbf{x}_{k+1} = A\mathbf{x}_k + \mathbf{e}_k, \tag{6.62}$$

$$\mathbf{e}_{k+1} = A\mathbf{e}_k, \qquad k = 0, .., \frac{N}{2} - 1, \tag{6.63}$$

where $R^{-1} = \frac{2}{N} \in \mathbb{R}^{p\times p}$, and $Q_0^{-1} = qI \in \mathbb{R}^{n\times n}$. As in the experiments of Chapter 5, the matrices R^{-1} give equal weight to all observations, and are not supposed to represent error covariances. The value of q is sometimes taken to be zero, in which case we do not constrain the size of the correction term to be small.

In the experiments, assimilation is carried out on the interval $t \in [0, \frac{1}{2}]$ over which observations are available. The solution at time $t = \frac{1}{2}$ is then used to initiate a forecast over the interval $[\frac{1}{2}, 1]$.

Starting from the correct initial conditions, and using $\mu = \frac{1}{2}$, the upwind scheme (typically of schemes which are first order accurate) exhibits numerical dissipation, which smears shocks. This model error becomes less severe as the grid is refined (keeping $\mu = \frac{1}{2}$). The aim of the experiments is to compare the performance of different control vectors in compensating for model error during the assimilation interval. We further investigate whether the assimilation produces an improvement in the subsequent forecast in each case. We investigate the following cases using the different control vectors.

Case a) Imperfect model, known initial state

The performance of assimilation using the initial state and the evolving correction term as control vectors is compared for different values of q and p. We also see

how performing data assimilation using the evolving correction term as a control vector compares with reducing model error by increasing the spatial and temporal resolution of the model.

Case b) Imperfect model, unknown initial state

The performance of assimilation using the initial state, the evolving correction term and both together as control vectors is compared for different values of q and p.

6.5 Results

The figures referred to in the text can be found at the end of this section. In each case the impact of the assimilation may be judged by comparing the solution with assimilation (dashed line) to the background solution (dotted line) in which no assimilation is performed.

6.5.1 Case a): Imperfect model, known initial state

Using the initial state as a control vector

When the full set of 40 observations are used, the initial state recovered is close to the true initial state, except that peaks are introduced at the corners of the square wave, as Fig. 6.1 shows. Hence, the impact of the numerical dissipation which smears down the corners, is less at later times in the assimilation. As noted in Chapter 5, when the initial state is used as a control vector in the presence of model error, the solution is closest to the true solution in the middle of the assimilation interval. Using the initial state as the control vector modifies the impact of model error by distributing its effects throughout the assimilation interval. At the the end of the assimilation interval, the solution is closer to the true solution than the model run started from true initial state, and as a result, the forecast remains slightly closer to the true model state.

If 20 observations are used at every timestep of the assimilation interval (Fig. 6.2), the results are similar to when the full set of observations is used, except that in

this case, as well as having peaks at the corners, the initial state also contains erroneous spikes away from the observation points, which are soon smoothed away as the solution evolves. As discussed in the context of the experiments of Chapter 5, the minimization is not sensitive to these errors in the initial state. When fewer than 20 observations are used, these spikes in the recovered initial state are larger, although the solution at later times is still good. This is illustrated in Fig. 6.3 which shows the case when $p = 5$. Even though the impact of this erroneous initial state is soon eliminated, it is clearly not a desirable solution, and should be treated either by including a background estimate of the initial state, or by imposing some other smoothness condition.

Using an evolving correction term as the control vector

The results using a full set of observations with $q = 10$ are shown in Fig. 6.4. Convergence is achieved in 25 iterations in this case. With $q = 0$ the results look very similar, but the stopping criterion had not been satisfied when the minimiziation was stopped after 100 iterations. As found in the experiments of Chapter 5, convergence occurs more quickly using larger values of q. Increasing the value of q to 100 produces visibly less accurate results, however.

Fig. 6.4 shows how the correction term compensates well for the effects of model error, and produces a solution which is better than the background solution starting from the true initial state, throughout the assimilation interval. Compared with the solution produced using the initial state as the control vector, the reduction in model error in the middle and at the end of the assimilation interval is achieved to a similar extent, but this time there is no corresponding increase in model error at the beginning of the assimilation interval.

There are two other advantages in this approach. Firstly, it provides a way of improving a subsequent forecast by including the evolving correction term in the forecast. Fig. 6.4 shows a considerable improvement over the original solution during the forecast period, although this does involve the extra cost of evolving the correction term as well as the full model state. The second advantage is that the solution obtained behaves well even when fewer observations are used, and there are

no erroneous spikes in the data sparse areas. Fig. 6.5 shows the good results obtained using 20 observations, and $q = 10$ in 22 iterations. When only 5 observations are used, the results are still good, as Fig. 6.6 shows.

In this example it is not possible to benefit by reducing the dimension of the correction term, (by using $m < n$ as in the examples of Chapter 5), because the wave and hence the impact of the model error travels across the whole of the model domain on the time interval we consider. In other applications, however, in which we may wish to treat the impact of numerical model error near a discontinuity which only travels over part of the model domain during the assimilation interval, it would be possible to reduce the dimension of the correction term so that it influences this area only.

The results of these experiments show that, if the initial state is known to a good enough approximation, using the evolving correction term as a control vector provides a better way of dealing with this type of model error than using the initial state, although the amount of work needed at each iteration in is approximately doubled.

In these experiments, model error is due to lack of resolution, and a more efficient way of correcting for this type of model error is of course to increase the resolution. We compare the results using the evolving correction term with the model solution at twice the spatial and temporal resolution, keeping μ fixed, ie using $\Delta z = \frac{1}{80}$, $\Delta t = \frac{1}{160}$. The results of these experiments are shown in Fig. 6.7 (using the full set of observations) and in Fig. 6.8 (using 20 observations). During the assimilation interval, the quality of the solution using the evolving correction term is very similar to the solution (without assimilation) at double spatial and temporal resolution. The forecast using the evolving correction term is slightly better than the forecast using double resolution.

Carrying out data assimilation using the evolving correction term to correct for model errors due lack of resolution involves more work than simply increasing the model resolution. However, these experiments indicate that the evolving correction term could be efficient in compensating for model error which travels with the model solution.

6.5.2 Case b): Imperfect model, unknown initial state

Using the initial state as a control vector

The results using the initial state as the control vector are the same whether the true initial state is known or not, but the minimization procedure requires a few more iterations if it is not known.

Using an evolving correction term as the control vector

If the first guess of the unknown initial state is taken to be zero, the evolving correction term must make up for very large errors. Fig. 6.9 shows the results using the full set of observations, and Fig. 6.10 shows the results using $p = 20$. The solution produced is the right shape, under-estimating the true solution in the first half of the assimilation interval, and over-estimating it in the second half. In these experiments it is not appropriate to include the evolving correction term in the forecast period.

Using both control vectors

Fig. 6.11 shows the solution produced using the full set of observations and $q = 1$. However, the stopping criterion had not been reached when the minimization was terminated after 100 iterations. The solution improves on the solution started from the true initial state, and on the solution obtained using the initial state only as the control vector. Using the evolving correction term in the forecast reduces the effect of model error as before. If the value of q is increased to 10, the initial state control vector appears to have too much influence, and the solution is less accurate at the initial time. Using larger values of q also results in a deterioration of the quality of the forecast, and does not succeed in reducing the number of iterations of descent algorithm to less than 100. Fig. 6.12 shows that fairly good results are still achieved using 20 observations. Fig. 6.13 shows that the results using only 5 observations are poorer, but still an improvement on the background solution.

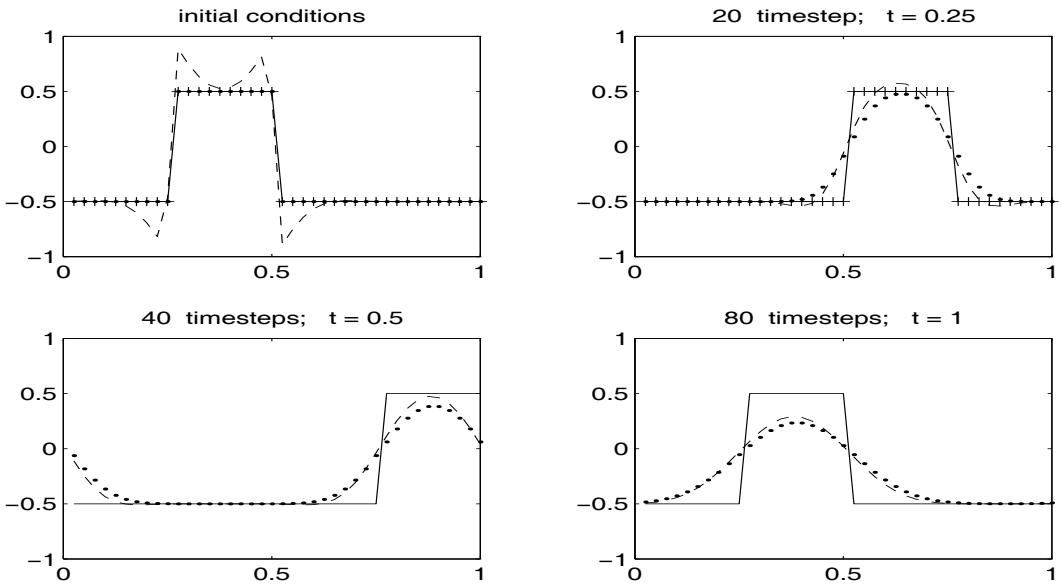

Figure 6.1: Variational assimilation using the initial state as the control vector. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 40 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

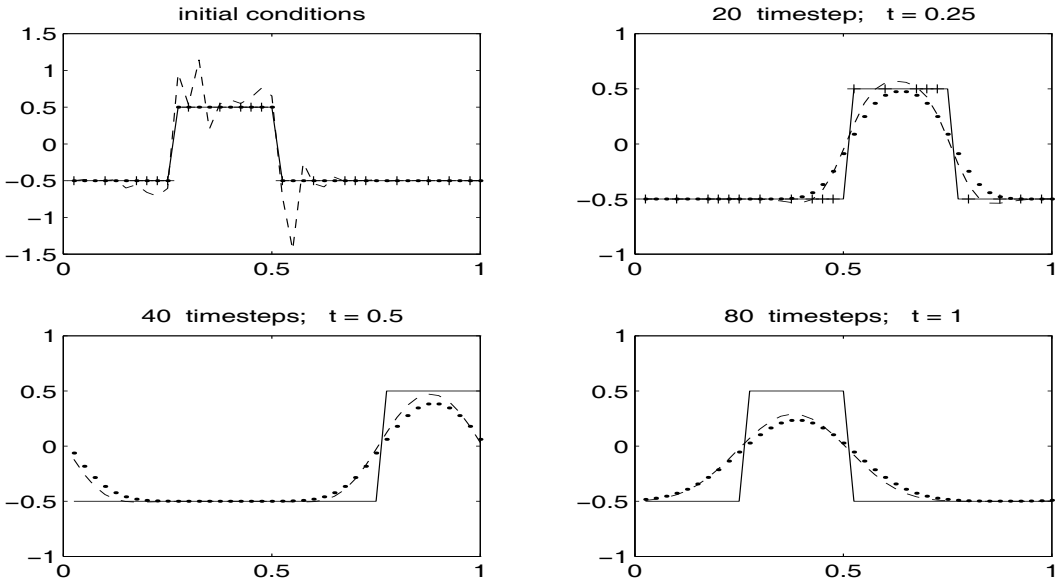

Figure 6.2: As Fig. 6.1, but using 20 observations.

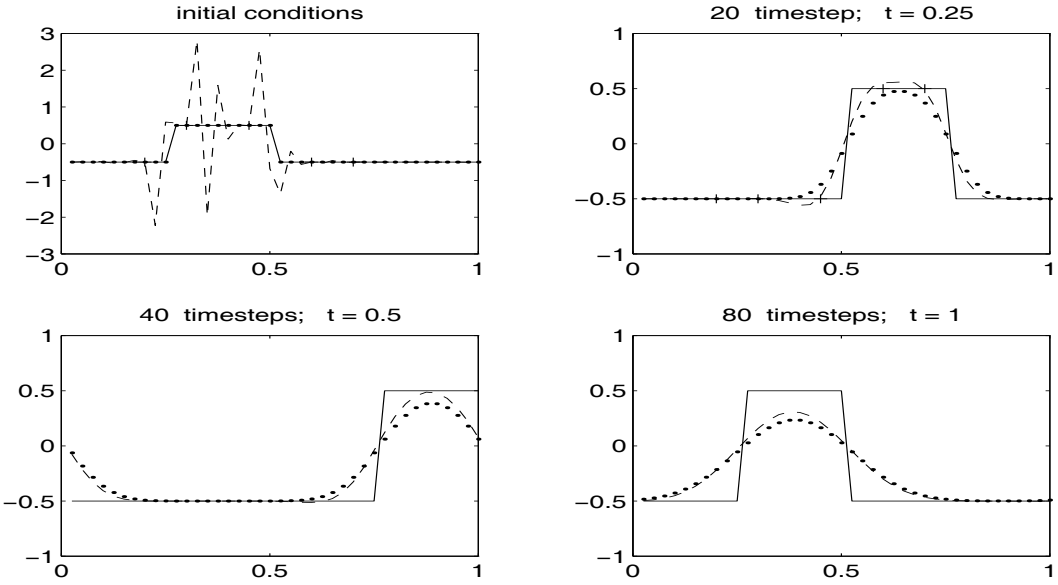

Figure 6.3: As Fig. 6.1, but using 5 observations.

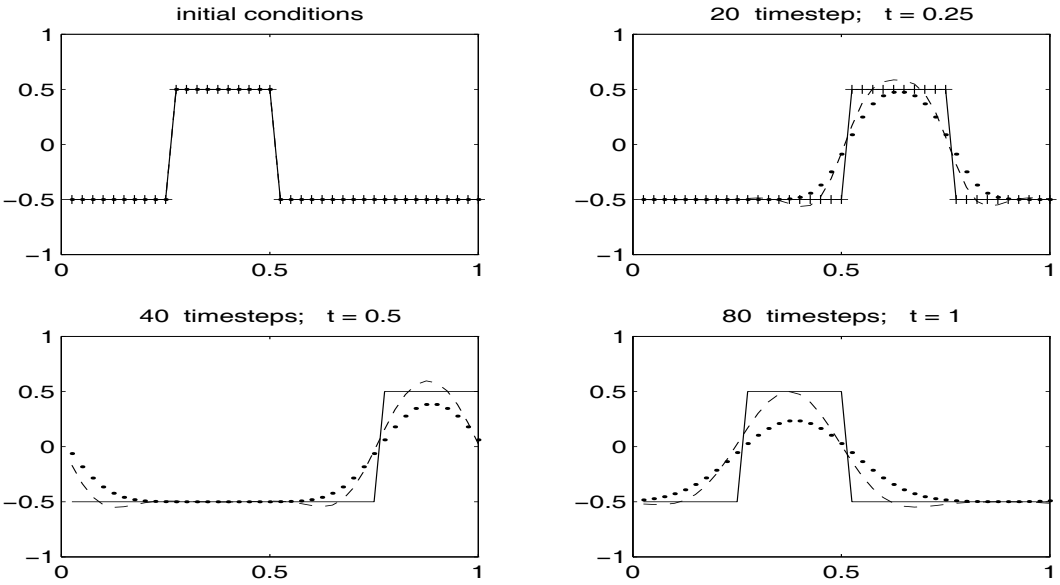

Figure 6.4: Variational assimilation using the evolving correction term as the control vector, with $q = 10$. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 40 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

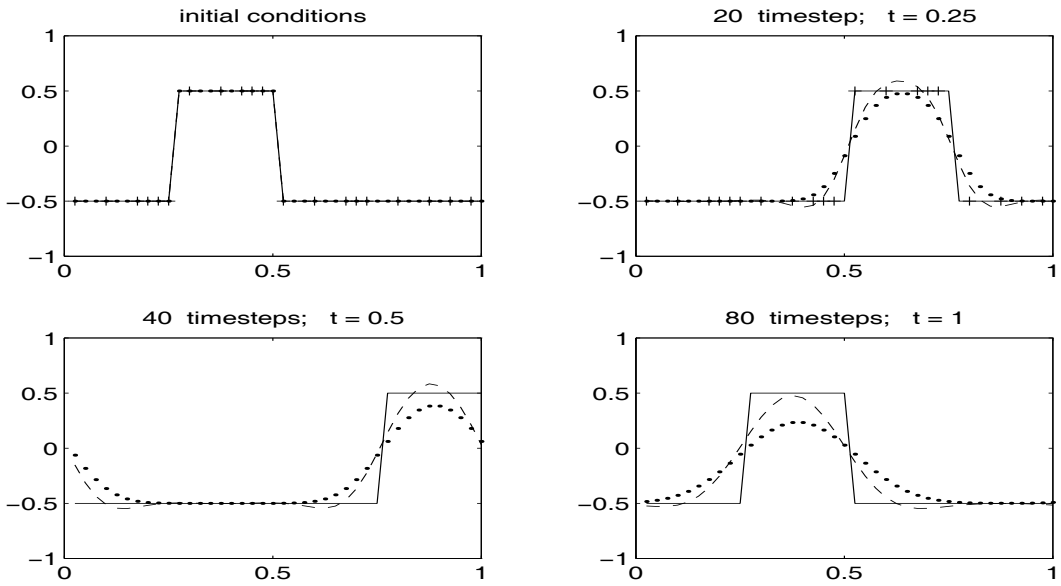

Figure 6.5: As Fig. 6.4, but using 20 observations.

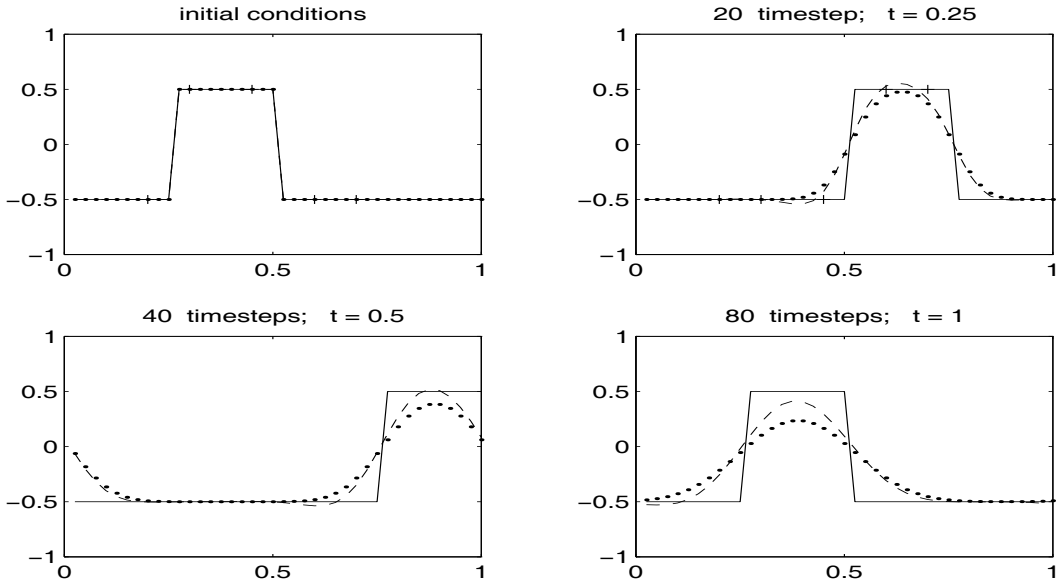

Figure 6.6: As Fig. 6.4, but using 5 observations.

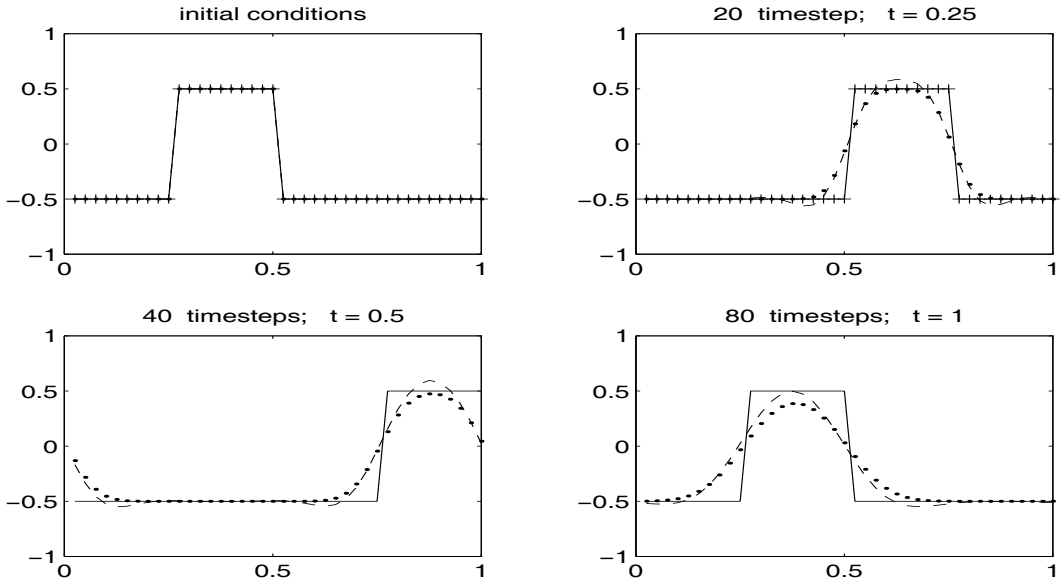

Figure 6.7: As Fig. 6.4, but this time the background solution (no assimilation) is performed at twice the resolution.

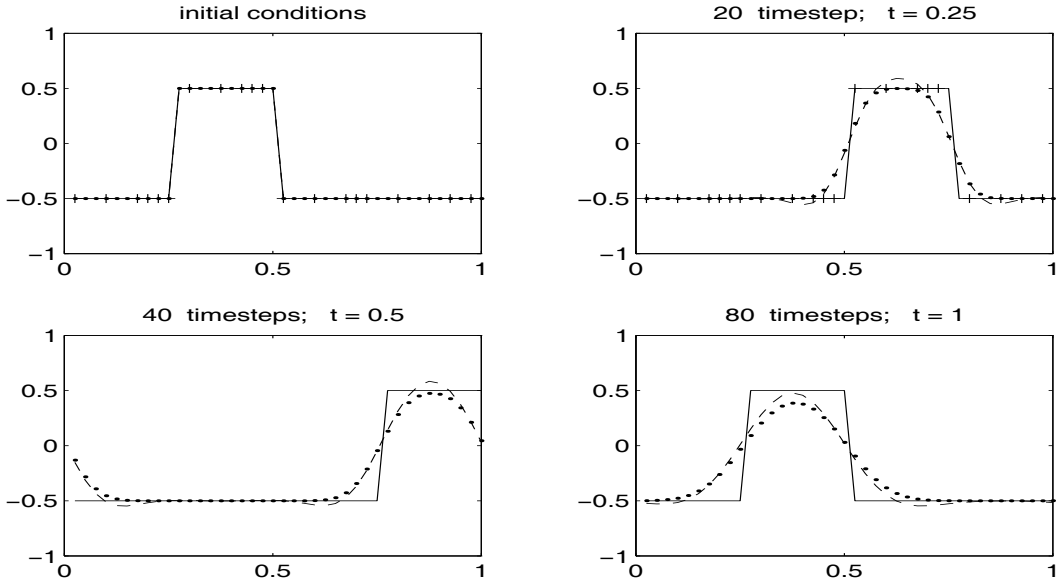

Figure 6.8: As Fig. 6.7, but using 20 observations.

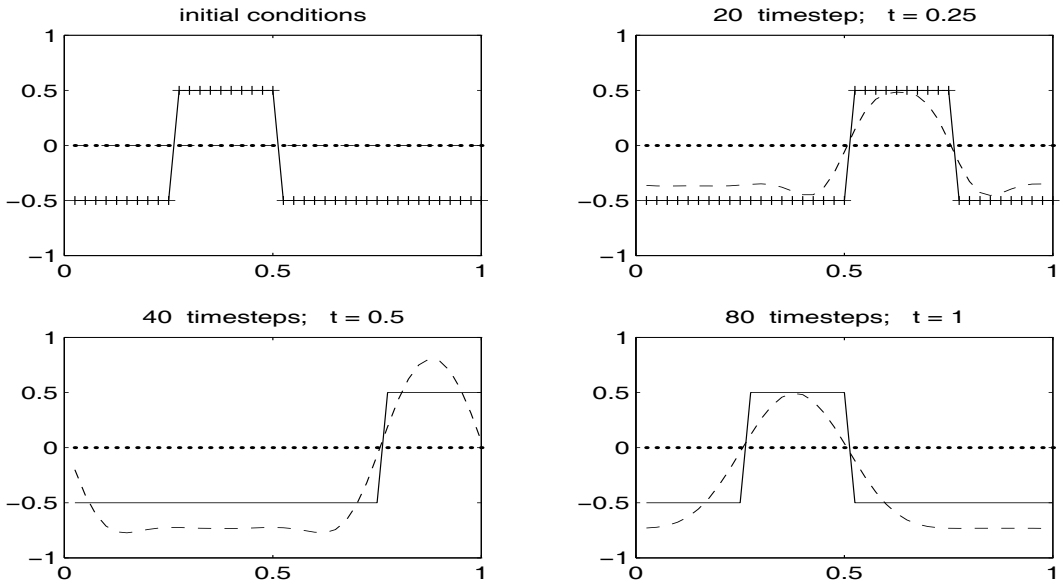

Figure 6.9: Variational assimilation using the evolving correction term as the control vector, with $q = 1$. In this case the true initial state is unknown. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 40 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

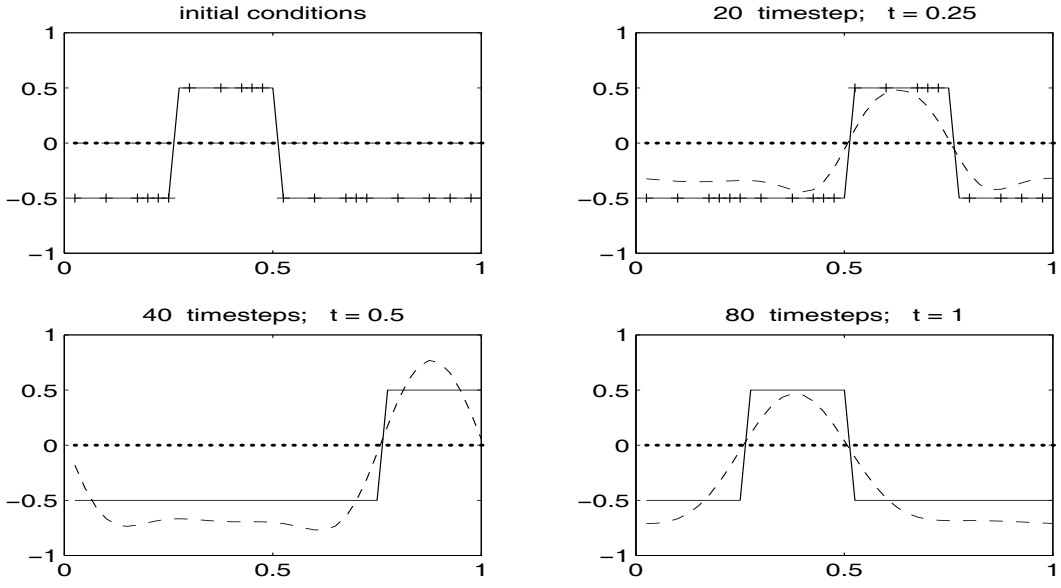

Figure 6.10: As Fig. 6.9, but using 20 observations.

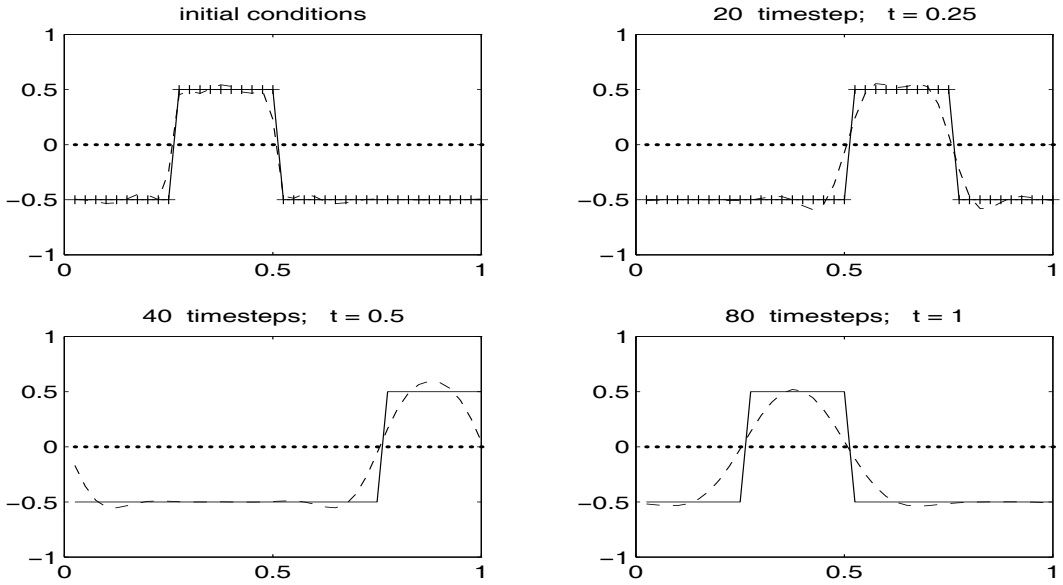

Figure 6.11: Variational assimilation using both the initial state and the evolving correction term as control vectors, with $q = 1$. In this case the true initial state is unknown. Assimilation on the interval $t \in [0, \frac{1}{2}]$ using 40 observations, followed by a forecast on the interval $t \in [\frac{1}{2}, 1]$. Solid line: true solution; dotted line: background solution (no assimilation); dashed line: solution with assimilation; crosses: observations.

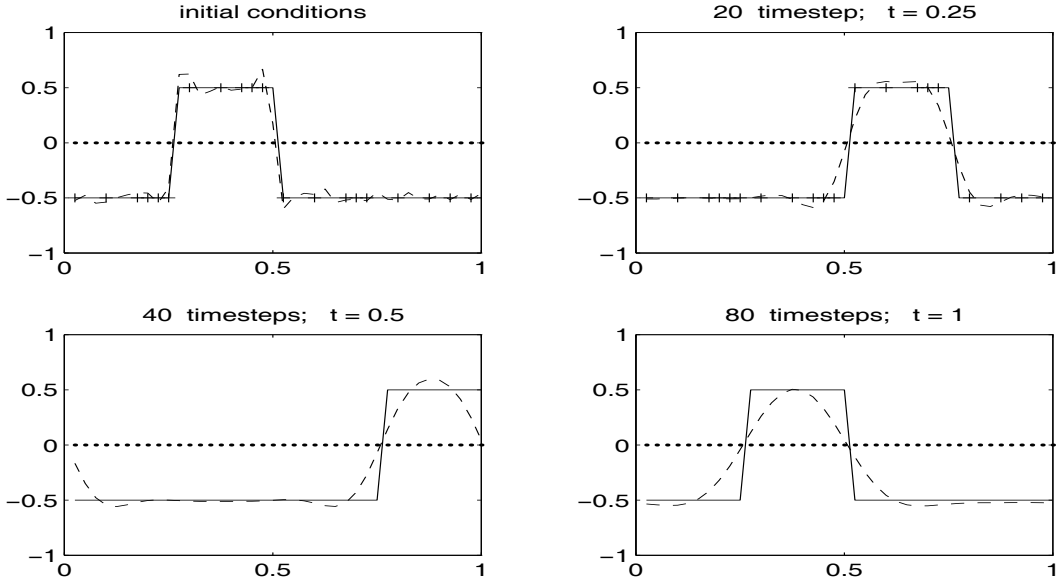

Figure 6.12: As Fig. 6.11, but using 20 observations.

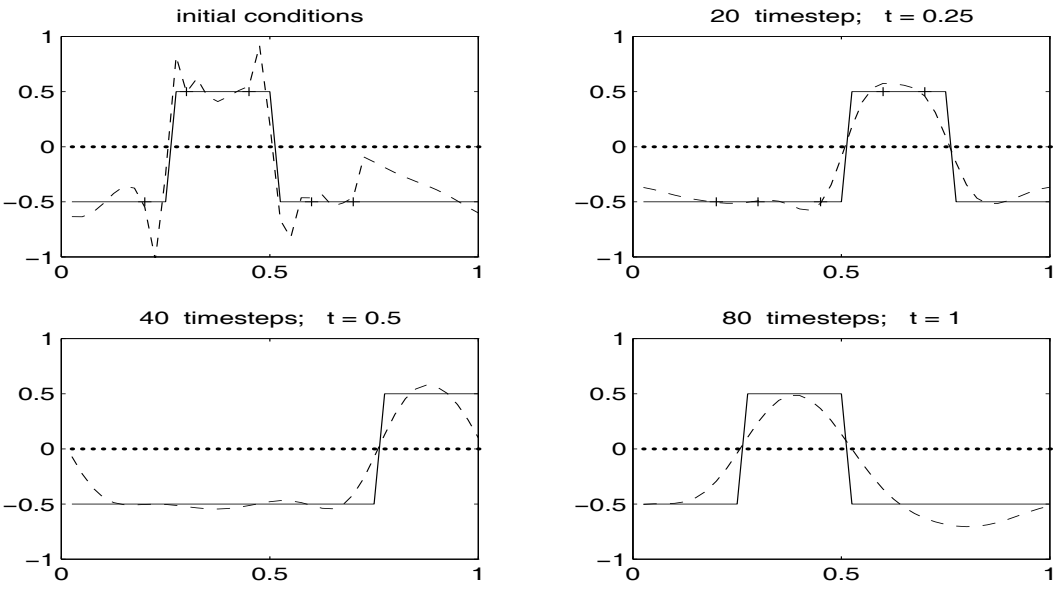

Figure 6.13: As Fig. 6.11, but using 5 observations.

6.6 Summary and conclusions

Summary of the theory

We began this chapter with a discussion on why it is important to account for model error in data assimilation, and on some of the limitations of the approach taken to model error in the standard Kalman filter. We then considered a general representation of model error made up of serially correlated and serially uncorrelated components, and gave examples of different representations of model error which might be suitable in different situations. We suggested that the technique of state augmentation could be used in data assimilation to estimate the serially correlated components of model error along with the model state. This leads to a generalization of the least squares problem of Chapter 4 to deal with serially correlated model error.

The correction term technique can be interpreted as giving an optimal solution to this general data assimilation problem in the case that model error is a constant bias error with no sequentially uncorrelated component. We also suggested a generalization of the correction term technique to allow for model error that changes with the state evolution, or to allow for more general forms of model error by including more than one correction term.

Conclusions from experiments with the evolving correction term

In Section 6.4 we used the generalized correction term technique with an "evolving correction term". We applied this to the linear advection equation with the upwind scheme discretization, an example in which the model error is numerical dissipation. For this example, using a constant correction term does not correct for the effects of model error at all. Using the initial state as the control vector, however, does compensate to some extent for the effects of model error. In this case, an initial state is found that over-exaggerates the corners of the wave, which to some extent compensates for the effects of the dissipation later in the assimilation interval. Although the solution at the end of the assimilation interval has been slightly improved by the assimilation, the benefits of this improvement are only very small by the end of a subsequent forecast interval.

Using the evolving correction term as a control vector compensates for the effects of model error better than using the initial state as the control vector. In this case, the evolving correction term compensates for the effects of model error throughout the assimilation interval, and also gives a improvement in the subsequent forecast. Another advantage of using the evolving correction term as the control vector is that the solutions produced are still smooth when fewer observations are used. The evolving correction term can also compensate to some extent for a wrong initial state, giving the best solution in the middle of the assimilation interval. In this case, however, it is not appropriate to include the evolving correction term in a subsequent forecast. Using both the initial state and the evolving correction term as control vectors however compensates very well for unknown initial state and model error during an assimilation interval, and for the effects of model error in a subsequent forecast, but the number of iterations required in this case remains high.

In this example, the effects of model error could more efficiently be corrected by refining the resolution of the model than by performing data assimilation using the evolving correction term technique. However, these simple experiments have shown that the evolving correction term technique could be used to compensate for the effects of model error which are likely to change with the model evolution.

Chapter 7

Experiments with a shallow water model

Here we describe experiments using a 1D nonlinear shallow water model which includes topography and rotation. In these experiments we aim to investigate to what extent the conclusions of the experiments in Chapters 5 and 6 hold in the context of more complex dynamics. We compare using the initial state, a constant correction term and both together as control vectors. In particular, we aim to see whether the constant correction term can compensate for model error on a significant timescale, when the model error depends on the model state, and hence changes in time. Further, since the correction term technique involves changing the model equations by adding on the correction term, we want to check that the correction term produced in the assimilation does in fact represent an approximation of model error. These experiments are carried out in an idealized context with a full set of observations which are not corrupted by noise. We also begin to look at the situation in which fewer, noisy observations are available. Finally, we check whether assimilation using the correction term technique can result in a better forecast than assimilation using the initial state as the control variable. This is important to check, because in Wergen's study [87] using the correction term technique produced good results during an assimilation interval, but had a detrimental impact on the ensuing forecast. A briefer description of the results from these experiments has been published in [41].

7.1 The shallow water model

The shallow water equations are often used in test problems in meteorology and oceanography because they describe flow which exhibits several features present in the flows of atmospheres and oceans. We consider the one-dimensional shallow water equations including rotation and bottom topography, which are given by

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{\partial \phi}{\partial x} = fv - g\frac{\partial H}{\partial x}, \tag{7.1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} = -fu, \tag{7.2}$$

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} + \phi\frac{\partial u}{\partial x} = 0, \tag{7.3}$$

where x and t represent the spatial and temporal independent variables, $u = u(x,t)$ and $v = v(x,t)$ are the eastward and northward components of velocity, and $\phi = \phi(x,t)$ is the geopotential given by $\phi = g\eta(x,t)$ where g is the acceleration due to gravity and $\eta(x,t)$ is the depth of the fluid. The height of the bottom topography is represented by $H = H(x)$, and f is the Coriolis parameter. Periodic boundary conditions are assumed. The model equations are nonlinear, and describe flow which may develop hydraulic jumps.

The discretization we use is a finite difference discretization developed by Parrett and Cullen [69], and we refer to it as the PC scheme. It was developed to give a good representation of hydraulic jumps. Artificial diffusion is added to the model equations to eliminate the spurious oscillations which are generated by second order finite difference schemes near jumps. Also, the discretization is carried out on the model in flux form, since a non-conservative version of the same discretization was found to produce errors in the position and amplitude of the jumps.

Hence, the discretization is carried out on (7.1)-(7.3) written in flux form with artificial diffusion added,

$$\frac{\partial(\phi u)}{\partial t} + \frac{\partial(\phi u^2 + \frac{1}{2}\phi^2)}{\partial x} = f\phi v - g(\frac{\partial H}{\partial x})\phi + K\frac{\partial^2(\phi u)}{\partial x^2}, \tag{7.4}$$

$$\frac{\partial(\phi v)}{\partial t} + \frac{\partial(\phi uv)}{\partial x} = -f(\phi u) + K\frac{\partial^2(\phi v)}{\partial x^2}, \tag{7.5}$$

$$\frac{\partial \phi}{\partial t} + \frac{\partial(\phi u)}{\partial x} = K\frac{\partial^2 \phi}{\partial x^2}, \tag{7.6}$$

for $x \in [0, 2\pi L]$, $t \in [0, T]$. The discretization is carried out with $\Delta x = \frac{1}{J}$, $\Delta t = \frac{1}{N}$, with discrete variables approximating the continuous variables as follows,

$$\phi_j^k \approx \phi(j\Delta x, k\Delta t), \quad u_j^k \approx u(j\Delta x, k\Delta t), \quad v_j^k \approx v(j\Delta x, k\Delta t), \tag{7.7}$$

for $k = 0, .., N$, $j = 0, .., J-1$. The discretization uses centred time and space differencing, except for the diffusion terms, in which forward time differencing is used for stability. The discrete model is

$$\begin{aligned} m_j^{k+1} = m_j^{k-1} & -\frac{\Delta t}{2\Delta x}\{(u_{j+1}^k + u_j^k)(m_{j+1}^k + m_j^k) - (u_j^k + u_{j-1}^k)(m_j^k + m_{j-1}^k) \\ & +((\phi_{j+1}^k)^2 - (\phi_{j-1}^k)^2)\} \\ & -g\frac{\Delta t}{2\Delta x}\{(\phi_{j+1}^k + \phi_j^k)(H_{j+1} - H_j) + (\phi_j^k + \phi_{j-1}^k)(H_j - H_{j-1})\} \\ & +2\Delta t f n_j^k + 2\frac{\Delta t}{\Delta x^2}K(m_{j+1}^{k-1} - 2m_j^{k-1} + m_{j-1}^{k-1}) \end{aligned} \tag{7.8}$$

$$\begin{aligned} n_j^{k+1} = n_j^{k-1} & -\frac{\Delta t}{2\Delta x}\{(v_{j+1}^k + v_j^k)(m_{j+1}^k + m_j^k) - (v_j^k + v_{j-1}^k)(m_j^k + m_{j-1}^k)\} \\ & -2\Delta t f m_j^k + 2\frac{\Delta t}{\Delta x^2}K(n_{j+1}^{k-1} - 2n_j^{k-1} + n_{j-1}^{k-1}) \end{aligned} \tag{7.9}$$

$$\phi_j^{k+1} = \phi_j^{k-1} \quad -\frac{\Delta t}{\Delta x}(m_{j+1}^k - m_{j-1}^k) + 2\frac{\Delta t}{\Delta x^2}K(\phi_{j+1}^{k-1} - 2\phi_j^{k-1} + \phi_{j-1}^{k-1}) \tag{7.10}$$

for $k = 1, .., N-1$, $j = 0, .., J-1$, with periodic boundary conditions, so

$$u_J^k = u_0^k, \quad v_J^k = v_0^k, \quad \phi_J^k = \phi_0^k, \qquad k = 0, ..N-1, \tag{7.11}$$

where

$$m_j^k = \phi_j^k u_j^k, \tag{7.12}$$

$$n_j^k = \phi_j^k v_j^k, \tag{7.13}$$

for $k = 0, .., N-1$, $j = 0, .., J-1$. The first time-step equations are specified using forward time differences,

$$\begin{aligned} m_j^1 = m_j^0 & -\frac{\Delta t}{2\Delta x}\{(u_{j+1}^0 + u_j^0)(m_{j+1}^0 + m_j^0) - (u_j^0 + u_{j-1}^0)(m_j^0 + m_{j-1}^0) \\ & +((\phi_{j+1}^0)^2 - (\phi_{j-1}^0)^2)\} \\ & -g\frac{\Delta t}{2\Delta x}\{(\phi_{j+1}^0 + \phi_j^0)(H_{j+1} - H_j) + (\phi_j^0 + \phi_{j-1}^0)(H_j - H_{j-1})\} \\ & +2\Delta t f n_j^0 + 2\frac{\Delta t}{\Delta x^2}K(m_{j+1}^0 - 2m_j^0 + m_{j-1}^0) \end{aligned} \tag{7.14}$$

$$n_j^1 = n_j^0 - \frac{\Delta t}{2\Delta x}\{(v_{j+1}^0 + v_j^0)(m_{j+1}^0 + m_j^0) - (v_j^0 + v_{j-1}^0)(m_j^0 + m_{j-1}^0)\}$$
$$-2\Delta t f m_j^0 + 2\frac{\Delta t}{\Delta x^2}K(n_{j+1}^0 - 2n_j^0 + n_{j-1}^0) \quad (7.15)$$

$$\phi_j^1 = \phi_j^0 - \frac{\Delta t}{\Delta x}(m_{j+1}^0 - m_{j-1}^0) + 2\frac{\Delta t}{\Delta x^2}K(\phi_{j+1}^0 - 2\phi_j^0 + \phi_{j-1}^0) \quad (7.16)$$

where the initial conditions are to be specified.

Since discrete models which include artificial diffusion do not always converge to the correct solution of (7.1)-(7.3), the PC scheme was compared in [69] with a method (Glimm's method) which has been proved to converge to the physically correct solution. The PC scheme was found to give good agreement to corresponding solutions of Glimm's method for several test cases involving hydraulic jumps.

After coding up this model in Fortran 77, we tested it by comparing results with those obtained in two of the examples given in [69]. We describe one of these examples here, since it was modified to provide the example to be used for our experiments. The Coriolis parameter is set at the value for 30^{O} North, ie $f = 7.292 \times 10^{-5}\mathrm{s}^{-1}$. We use a spatial discretization of 100 grid points, so $J = 100$. In the first experiment of the paper, there is no topography, but a hydraulic jump evolves for certain Rossby and Froude numbers from smooth initial conditions given by

$$u(x,0) = U\cos(x/L), \quad (7.17)$$
$$v(x,0) = 0, \quad (7.18)$$
$$\phi(x,0) = \phi_m + U\{(U/8)\cos(2x/L) + (\phi_m - U^2/8)^{\frac{1}{2}}\cos(x/L)\}, \quad (7.19)$$

where the length of the domain is $2\pi L$, $\phi_m = g\eta_m$ where η_m is the mean depth of the fluid, and U is a constant. In our case, we ensured the required Rossby number $Ro = U/fL = 1$ and Froude number $F = U/\phi_m^{\frac{1}{2}} = 1$ were satisfied by choosing the constants L, U and ϕ_m as

$$U = 1\mathrm{ms}^{-1}, \quad (7.20)$$
$$\phi_m = 1\mathrm{m}^2\mathrm{s}^{-2}, \quad (7.21)$$
$$L = 7.292 \times 10^5\mathrm{m}. \quad (7.22)$$

Since $\Delta x = 2\pi L/J$, we have $\Delta x = 4.58 \times 10^4$m, or approximately 46km. We chose a timestep to satisfy $\frac{\Delta t}{\Delta x} = \frac{1}{10}$, so $\Delta t = 4.58 \times 10^3$s (which is approximately one hour 15 minutes). As in the paper [69], we chose $K = 2.5 \times 10^4 \mathrm{m}^2\mathrm{s}^{-1}$.

Results from this test case were plotted in non-dimensional form, and seen to give good agreement with the corresponding figures in [69]. Our model was also tested on the examples given in [69] which include topography, and found to agree with the results in the paper in these cases, too.

7.2 The data assimilation problem

We define the model state $\mathbf{x}_k \in \mathbb{R}^{3J}$ at time t_k to be the vector

$$\mathbf{x}_k = (m_0^k, .., m_{J-1}^k, n_0^k, .., n_{J-1}^k, \phi_0^k, .., \phi_{J-1}^k)^T, \tag{7.23}$$

and we define the correction term to be the vector $\mathbf{e} \in \mathbb{R}^{3J}$ given by

$$\mathbf{e} = (e_0^{(m)}, .., e_{J-1}^{(m)}, e_0^{(n)}, .., e_{J-1}^{(n)}, e_0^{(\phi)}, .., e_{J-1}^{(\phi)})^T. \tag{7.24}$$

We suppose that we have observations over timesteps t_0 to t_{N-1}. The data assimilation problem we address is to minimize

$$\mathcal{J} = \mathcal{J}_o + \frac{1}{2}\mathbf{e}Q^{-1}\mathbf{e} \tag{7.25}$$

with respect to the control vector or vectors being used, subject to the constraint that the model equations (7.8)-(7.16) hold, where the observational part of the cost function $\mathcal{J}_o$ is to be specified later. The matrix Q^{-1} is given by qI where I is the identity matrix, and different values of q are used in the experiments. We suppose that there are a large number of observations, and so do not include a background of the initial state in the cost function.

7.2.1 The adjoint model

We wish to minimize the cost function $\mathcal{J}$ subject to each of the model equations (7.8)-(7.10) with (7.14)-(7.16) and the relations (7.12),(7.13), and we introduce a Lagrange multiplier for each of these model equations.

As usual, we let $\mathcal{L}$ denote the Lagrangian function associated with $\mathcal{J}$. Hence $\mathcal{L}$ is made up of $\mathcal{J}$ plus the sum of all the Lagrange multipliers multiplying their respective model constraints. We let λ_j^k multiply (7.8), p_j^k multiply (7.9) and μ_j^k multiply (7.10) for $k = 1, .., N-1$, $j = 0, .., J-1$. We let λ_j^0 multiply (7.14), p_j^0 multiply (7.15) and μ_j^0 multiply (7.16) for $j = 0, .., J-1$, and finally we let ν_j^k multiply (7.12) and ρ_j^k multiply (7.13) for $k = 0, .., N-1$, $j = 0, .., J-1$. We assume that the boundary conditions (7.11) are substituted directly into the model equations.

The adjoint equations are given by

$$\begin{aligned}\lambda_j^{k-1} = \lambda_j^{k+1} & -\frac{\Delta t}{2\Delta x}\{(u_{j+1}^{k-1}+u_{j-1}^{k-1})(\lambda_j^k-\lambda_{j+1}^k)-(u_j^{k-1}+u_{j-1}^{k-1})(\lambda_j^k-\lambda_{j-1}^k) \\ & +2(\mu_{j-1}^k-\mu_{j+1}^k)\} \\ & -\frac{\Delta t}{2\Delta x}\{(v_{j+1}^{k-1}+v_{j-1}^{k-1})(p_j^k-p_{j+1}^k)-(v_j^{k-1}+v_{j-1}^{k-1})(p_j^k-p_{j-1}^k)\} \\ & -2\Delta t f p_j^k + 2\frac{\Delta t}{\Delta x^2}K(\lambda_{j-1}^{k+1}-2\lambda_j^{k+1}+\lambda_{j+1}^{k+1})-\nu_j^k \\ & +\frac{\partial \mathcal{J}_o}{\partial m_j^{k-1}}, \end{aligned} \tag{7.26}$$

$$\begin{aligned} p_j^{k-1} = & \; p_j^{k+1}+2\Delta t f\lambda_j^k+2\frac{\Delta t}{\Delta x^2}K(p_{j-1}^{k+1}-2p_j^{k+1}+p_{j+1}^{k+1})-\rho_j^k \\ & +\frac{\partial \mathcal{J}_o}{\partial n_j^{k-1}}, \end{aligned} \tag{7.27}$$

$$\begin{aligned}\mu_j^{k-1} = \mu_j^{k+1} & -\frac{\Delta t}{\Delta x}(\lambda_{j-1}^k-\lambda_{j+1}^k)\phi_j^{k-1} \\ & -\frac{\Delta t}{2\Delta x}g\{(\lambda_{j-1}^k+\lambda_j^k)(H_j-H_{j-1})+(\lambda_j^k+\lambda_{j+1}^k)(H_{j+1}-H_j)\} \\ & +2\frac{\Delta t}{\Delta x^2}K(\mu_{j-1}^{k+1}-2\mu_j^{k+1}+\mu_{j+1}^{k+1})-\nu_j^k u_j^{k-1}+\rho_j^k v_j^{k-1} \\ & +\frac{\partial \mathcal{J}_o}{\partial \phi_j^{k-1}}, \end{aligned} \tag{7.28}$$

$$\nu_j^k = \frac{\Delta t}{2\Delta x\phi_j^{k-1}}\{(m_{j+1}^{k-1}+m_j^{k-1})(\lambda_j^k-\lambda_{j+1}^k)-(m_j^{k-1}+m_j^{k-1})(\lambda_j^k-\lambda_{j-1}^k)\}, \tag{7.29}$$

$$\rho_j^k = \frac{\Delta t}{2\Delta x\phi_j^{k-1}}\{(m_{j+1}^{k-1}+m_j^{k-1})(p_j^k-p_{j+1}^k)-(m_j^{k-1}+m_j^{k-1})(p_j^k-p_{j-1}^k)\}, \tag{7.30}$$

for $k = N, .., 1$, $j = 0, .., J-1$ with

$$\lambda_j^{N+1} = 0, \qquad p_j^{N+1} = 0, \qquad \mu_j^{N+1} = 0, \tag{7.31}$$

$$\lambda_j^{N} = 0, \qquad p_j^{N} = 0, \qquad \mu_j^{N} = 0, \tag{7.32}$$

for $j = 0, ..., J-1$.

7.2.2 The gradients of $\mathcal{L}$ with respect to the control vectors

The partial derivatives of the Lagrangian $\mathcal{L}$ with respect to the variables making up the initial state $\mathbf{x}_0$ are given by

$$\frac{\partial \mathcal{L}}{\partial m_j^0} = -\lambda_j^0 - \lambda_j^1 - \frac{\Delta t}{2\Delta x^2} K(\lambda_{j+1}^1 - 2\lambda_j^1 + \lambda_{j-1}^1), \tag{7.33}$$

$$\frac{\partial \mathcal{L}}{\partial n_j^0} = -p_j^0 - p_j^1 - \frac{\Delta t}{2\Delta x^2} K(p_{j+1}^1 - 2p_j^1 + p_{j-1}^1), \tag{7.34}$$

$$\frac{\partial \mathcal{L}}{\partial \phi_j^0} = -\mu_j^0 - \mu_j^1 - \frac{\Delta t}{2\Delta x^2} K(\mu_{j+1}^1 - 2\mu_j^1 + \mu_{j-1}^1). \tag{7.35}$$

Hence, the gradient of $\mathcal{L}$ with respect to $\mathbf{x}_0$ is

$$\nabla_{\mathbf{x}_0} \mathcal{L} = (\frac{\partial \mathcal{L}}{\partial m_0^0}, .., \frac{\partial \mathcal{L}}{\partial m_{J-1}^0}, \frac{\partial \mathcal{L}}{\partial n_0^0}, .., \frac{\partial \mathcal{L}}{\partial n_{J-1}^0}, \frac{\partial \mathcal{L}}{\partial \phi_0^0}, .., \frac{\partial \mathcal{L}}{\partial \phi_{J-1}^0})^T. \tag{7.36}$$

The partial derivatives of $\mathcal{L}$ with respect to the variables in the control vector $\mathbf{e}$ are

$$\frac{\partial \mathcal{L}}{\partial e_j^{(m)}} = Q^{-1} e_j^{(m)} - \sum_{k=1}^{N-1} \lambda_j^k, \tag{7.37}$$

$$\frac{\partial \mathcal{L}}{\partial e_j^{(n)}} = Q^{-1} e_j^{(n)} - \sum_{k=1}^{N-1} p_j^k, \tag{7.38}$$

$$\frac{\partial \mathcal{L}}{\partial e_j^{(\phi)}} = Q^{-1} e_j^{(\phi)} - \sum_{k=1}^{N-1} \mu_j^k, \tag{7.39}$$

and the gradient of $\mathcal{L}$ with respect to $\mathbf{e}$ is

$$\nabla_{\mathbf{e}} \mathcal{L} = (\frac{\partial \mathcal{L}}{\partial e_0^{(m)}}, .., \frac{\partial \mathcal{L}}{\partial e_{J-1}^{(m)}}, \frac{\partial \mathcal{L}}{\partial e_0^{(n)}}, .., \frac{\partial \mathcal{L}}{\partial e_{J-1}^{(n)}}, \frac{\partial \mathcal{L}}{\partial e_0^{(\phi)}}, .., \frac{\partial \mathcal{L}}{\partial e_{J-1}^{(\phi)}})^T. \tag{7.40}$$

7.3 Description of the experiments

7.3.1 The true model state

For our experiments, we suppose that the true model state is defined by a run of the model with certain parameters and initial conditions. We use the values of f, J, L, Δx, Δt, and K specified in Section 7.1, but this time we use different initial conditions and a non-zero bottom topography. This low spatial resolution was chosen so that the dimension of the control vector would not be too large. It was noticed, however, that when the model was run at twice the spatial resolution, the results were not significantly different. In Experiments 1 and 2, the model is run for 100 timesteps ($N = 100$), and we take the assimilation interval $[0, T]$ to represent 100 timesteps.

The bottom topography is as given in [69], by

$$H(x) = H_c(1 - (x - \frac{L}{2})^2/a^2) \qquad 0 \leq \ (x - \frac{L}{2}) \ \leq a, \tag{7.41}$$

where H_c is half the initial water depth. We take a to correspond to a length of ten grid points. The shape of the bottom topography is shown in Fig. 7.1.

We define the true model initial state to be given by a fluid depth of 1m, and zero velocities, so we have (taking $g = 10\text{ms}^{-2}$)

$$m_j^0 = 0\text{m}^3\text{s}^{-3}, \tag{7.42}$$

$$n_j^0 = 0\text{m}^3\text{s}^{-3}, \tag{7.43}$$

$$\phi_j^0 = 10\text{m}^2\text{s}^{-2}, \tag{7.44}$$

for $j = 0, ..., J-1$. From this initial state, motion is initiated as fluid flows down from the ridge in the centre of the domain. A wave travels in each direction across the domain. This is illustrated in Fig. 7.1 which shows the true solution at the initial time and also after 50 and after 100 timesteps.

7.3.2 Observations

In Experiments 1 and 3, we suppose that we have a full set of observations, ie, observations of all the model state variables for all 100 timesteps. These observations

are the same as the true model state. In Experiment 3, we also carry out experiments in which we suppose that observations are available only for the first 50 timesteps.

In Experiment 2, we use observations corrupted by unbiased and sequentially uncorrelated random errors. The errors in the ϕ-field are uniformly distributed between -0.5 and 0.5, and the errors in the m- and n-fields are uniformly distributed between -0.25 and 0.25. We also suppose fewer observations are available in the spatial domain. We suppose that observations are available at every second, fourth or tenth grid point.

If we let $\tilde{m}_j^k$, $\tilde{n}_j^k$ and $\tilde{\phi}_j^k$ denote the observations of the model state variables m_j^k, n_j^k and ϕ_j^k, the observational part of the cost function is given by

$$\mathcal{J}_o = \frac{1}{2} \sum_{k=0}^{N-1} \sum_{j=0}^{J-1} c_j (m_j^k - \tilde{m}_j^k)^2 + c_j (n_j^k - \tilde{n}_j^k)^2 + c_j (\phi_j^k - \tilde{\phi}_j^k)^2, \tag{7.45}$$

where $c_j = 1$ if there are observations at the j^{th} grid point, and is zero otherwise. The partial derivatives of $\mathcal{J}_o$ with respect to the state variables to be included in the adjoint equations are given by

$$\frac{\partial \mathcal{J}_o}{\partial m_j^k} = c_j (m_j^k - \tilde{m}_j^k), \tag{7.46}$$

$$\frac{\partial \mathcal{J}_o}{\partial n_j^k} = c_j (n_j^k - \tilde{n}_j^k), \tag{7.47}$$

$$\frac{\partial \mathcal{J}_o}{\partial \phi_j^k} = c_j (\phi_j^k - \tilde{\phi}_j^k), \tag{7.48}$$

for $k = 0, ..., N-1$, $j = 0, ..., J-1$.

7.3.3 Model error

We carry out experiments with an imperfect model, in which we introduce the following two sources of model error. Both these sources of model error are very severe; this is done for exaggerated results in our experimental setting.

Model error i) Omission of bottom topography

Here we suppose that we have a model which neglects the "true" bottom topography as defined in (7.41). The model error at the j^{th} grid point and k^{th} timestep is

therefore

$$\varepsilon_j^{(m)k} = g\frac{\Delta t}{2\Delta x}\{(\phi_{j+1}^k + \phi_j^k)(H_{j+1} - H_j) + (\phi_j^k + \phi_{j-1}^k)(H_j - H_{j-1})\}, \quad (7.49)$$

$$\varepsilon_j^{(n)k} = 0, \quad (7.50)$$

$$\varepsilon_j^{(\phi)k} = 0. \quad (7.51)$$

Clearly, this model error does depend on the model state, and so is not constant in time. No motion is initiated when this model is used with the true initial state defined in (7.42)-(7.44), as the background solution in Fig. 7.5 shows.

Model error ii) Omission of rotation

In this case the Coriolis parameter is taken to be zero, and the model error at the j^{th} grid point and k^{th} timestep is

$$\varepsilon_j^{(m)k} = 2\Delta t f n_j^k, \quad (7.52)$$

$$\varepsilon_j^{(n)k} = -2\Delta t f m_j^k, \quad (7.53)$$

$$\varepsilon_j^{(\phi)k} = 0. \quad (7.54)$$

Again, the model error depends on the state and hence changes in time. In a model run started from the true initial state (7.42)-(7.44) with this model error, the n-field remains zero, and small errors in the m- and ϕ-fields develop in time as the background solution in Fig. 7.9 shows.

7.3.4 The descent algorithm

The descent algorithm used is the INRIA limited-memory quasi-Newton minimization package n1qn3.f, which is described in Chapter 2, Section 2.4. The stopping criterion used is

$$\frac{\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u}^i)}{\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u}^1)} < epsg = 10^{-4}, \quad (7.55)$$

where $\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u}^i)$ is the gradient of $\mathcal{L}$ with respect to the control vector $\mathbf{u}$ on the i^{th} iteration, and $\nabla_{\mathbf{u}}\mathcal{L}(\mathbf{u}^1)$ is the gradient on the first iteration. If the stopping criterion is not satisfied in 300 iterations, the minimization is terminated anyway.

The number $\hat{m}$ of updates used in forming the inverse Hessian is 6; a value between 5 and 10 is suggested in the program documentation [34] to provide a

compromise between a better approximation of the inverse Hessian using a high value of $\hat{m}$, and lower CPU time with a low value of $\hat{m}$.

7.3.5 The experiments

Experiment 1

The aim of this experiment is to compare the performance of the different control vectors in the presence of different types of model error, and of error in the initial state. A full set of observations, uncorrupted by error, is used. The following cases are investigated.

Case a) Perfect model, unknown initial state

In this case there is no model error, but the true initial state (7.42)-(7.44) is unknown. The background estimate of the initial state is

$$m_j^0 = 0\mathrm{m}^3\mathrm{s}^{-3}, \tag{7.56}$$

$$n_j^0 = 0\mathrm{m}^3\mathrm{s}^{-3}, \tag{7.57}$$

$$\phi_j^0 = 15\mathrm{m}^2\mathrm{s}^{-2}, \tag{7.58}$$

for $j = 0, ..., J-1$.

Case b) Omission of topography, known initial state

In this case the model error is type i) above, but the true initial state is known.

Case c) Omission of rotation, known initial state

In this case the model error is type ii) above, but the true initial state is known.

Case d) Omission of topography and rotation, and unknown initial state

Here model error of type i) and type ii) is present, and the background estimate of the initial state is as given in (7.56)-(7.58).

Experiment 2

Experiment 2, Case b) is repeated using observations corrupted by observational noise, and using fewer observations, also corrupted by observational noise. The aim here is to check to what extent the conclusions of Experiment 1 still hold in this more realistic case, rather than to explore the impact of increasing or reducing the number of observations. The following cases are investigated.

Case e) Observations with random error

Experiment 1b is carried out using observations corrupted by noise as described in Subsection 7.3.2.

Case f) Fewer observations with random error

Experiment 1b is carried out using fewer observations corrupted by noise. We suppose that observations (of all the state variables) are available only at every fourth timestep.

Experiment 3

In Experiment 1 we compare the performance of the different control vectors in compensating for model error and error in the initial state over an assimilation interval. In Experiment 3 our aim is to test whether an improvement at the end of the assimilation interval leads to an improved forecast.

Here we suppose observations are available over an assimilation period of 50 timesteps. The assimilation is carried out using either initial state or the correction term as a control vector, and then a "forecast" is carried out over the remaining 50 timesteps. The same is carried out using an assimilation interval of 100 timesteps, and a forecast interval of 100 timesteps.

Case g) Omission of rotation, known initial state

In this case the model error is type i) above, but the true initial state at the beginning of the assimilation interval is known.

Case h) Omission of rotation, known initial state

In this case the model error is type ii) above, but the true initial state at the beginning of the assimilation interval is known.

7.4 Results from the experiments

7.4.1 Experiment 1: Comparing different control vectors

The figures referred to here can be found at the end of this section.

Case a) Perfect model, unknown initial sate

Fig. 7.1 shows the true solution, and the background solution started with the wrong initial state. Fig. 7.3 illustrates that the errors in the background solution are an over-estimated ϕ-field throughout, and also that the waves travel too fast across the domain. Fig. 7.2 shows the solutions generated using each of the control vectors, and Fig. 7.4 shows the errors in each of these solutions. The improvement given by the assimilation in each case can be judged by comparing the solutions of Fig. 7.2 with the true solution and the background solution of Fig. 7.1, and by comparing the errors after assimilation (Fig. 7.4) with the errors in the background solution (Fig. 7.3).

Using the initial state as the control vector, it is possible to perfectly reconstruct the true solution with a perfect model and a full set of perfect observations (Fig. 7.2, Fig. 7.4). These results are as we expected, and are as we also found in the experiments of Chapters 5 and 6. This was achieved using 37 iterations of the minimization algorithm.

Using the correction term as the control vector also gives a significant reduction in the errors (Fig. 7.4). The errors in the m- and n-fields have successfully been treated by the correction term, and the waves now travel at the correct speed. Fig. 7.17 shows the actual correction term found by the assimilation. As seen in the experiments of Chapter 5, the solution is closest to the true solution in the middle of the assimilation interval, and the correction is too large at the end of the

assimilation interval. The results shown are for $q = 1$. Using $q = 10$, the correction term found is smaller, and so in this case the solution is closest to the true solution at the end rather than in the middle of the assimilation interval. For larger values of q fewer iterations are needed, for $q = 1$, 63 iterations are needed and using $q = 10$, 39 iterations are needed, which is similar to the number of iterations required using the initial state as the control vector. When q is increased further, however, the results are much poorer. The points we make here on the impact of different values of q are consistent with the conclusions we made from the experiments of Chapter 5.

When both control vectors are used together, the results achieved depend very strongly on the value of q. Using $q = 0$, the results are very similar to those obtained using only the correction term as the control vector, and as q is increased, the results become more like those obtained using the initial state only. The results shown in Fig. 7.2 and Fig. 7.4 are for $q = 100$. As found in the experiments of Chapters 5 and 6, using both control vectors together requires many more iterations, in this case (for all values of q), around 270 iterations.

Case b) Omitted topography, known initial state

In this case, the imperfect model started from the correct initial state generates no motion at all (Fig. 7.5). Hence, the background errors (Fig. 7.7) are large in all model fields, and propagate from the centre of the model domain right to its edges on the timescale of the assimilation. Fig. 7.8 shows that assimilation using each of the control vectors makes a very significant reduction in this background error.

When the initial state is used as the control vector, the correct initial *height* profile is produced, and this compensates for the omission of the topography in the model. This height profile in the ensuing motion is also good, but the depth is wrong. However, the errors in the m- and n-fields are now small, and so using the initial state as the control vector compensates very well for the effects of model error in this respect. Here 27 iterations were required for these results.

Using the correction term as a control vector successfully compensates for the model error. As Fig. 7.8 shows, the errors in the m- and n-fields are almost eliminated, and the error in the ϕ-field is reduced significantly. As can be seen in

equation (7.49), the actual model error at each timestep depends on the m-field. Fig. 7.18 shows that the correction term derived in the assimilation is a correction to this field only, and so it is reasonable to assume that the correction term found in the assimilation does indeed represent the temporal average of model error. The errors still existing in the ϕ-field at the end of the assimliation interval are presumably due to the fact that this average does not perfectly represent the actual model error. However, it is significant that the correction term representing a temporal average of model error compensates for the real model error on the timescale of the assimilation, since on this timescale the effects of the model error propagate half way across the model domain in each direction.

The results shown are again with $q = 1$, and this requires 55 iterations. Increasing q to 10 reduces this number to 36 iterations, but the results are slightly less accurate in this case.

Using both control vectors together and $q = 0$ produces very good results. Fig. 7.18 shows that in this case the correction term found in the assimilation is very much like that found when the correction term is the only control vector, but slightly smaller. Further, the initial state found in the assimilation is slightly different to the true initial state, and this has the effect of further reducing the error in the ϕ-field at the end of the assimilation interval. If the value of q is increased, the solution becomes more like that where only the initial state is used, and so not as good. Using only the correction term as the control vector works very well, but using the initial state as well it is possible to further compensate for the effects of model error. As before, the number of iterations using both control vectors is high; in this case 273 iterations are needed.

Case c) Omitted rotation, known initial state

The background solution of Fig. 7.9 illustrates that with the Coriolis parameter set at zero, no motion is initiated in the n-field. Fig. 7.11 shows the resulting errors in the n-field, and also shows very small errors in the m- and ϕ-fields by the end of the assimilation interval.

When the initial state is used as the control vector, the solution is closest to the

true solution in the middle of the time interval. A "wrong" initial state is found in the assimilation procedure, but the ensuing solution is closer to the true solution in the middle and at the end of the assimilation interval. This is typical of the way that assimilation using the initial state as the control vector compensates for model error, as we saw in the experiments of Chapters 5 and 6. In this case, 29 iterations were performed.

Using the correction term as the control vector reduces the errors more than using the initial state as the control vector in the middle and at the end of the assimilation, as Fig. 7.12 shows. The figures show the results using $q = 1$, and in this case 30 iterations are needed. Again, for larger values of q the results are less accurate. Fig. 7.19 shows that the correction term found in the assimilation corrects the n-field. This is appropriate since the background error (Fig. 7.10) is restricted to this field. Therefore it seems reasonable to assume that the correction term found in the assimilation does represent a time average of the model error.

When both control vectors are used together with $q = 0.2$, the errors in the middle and at the end of the assimilation interval are slightly smaller than using either of the control vectors on their own. In this case, using $q = 0$ produces a correction term which seems to include a spurious correction to the m-field, although this does not affect the results. Using $q = 0.2$ produces a correction term which is very similar to that produced using only the correction term as a control vector, as Fig. 7.19 shows. Again, the number of iterations required using both control vectors is high, 250 iterations in this case.

Case d) Omitted topography and rotation, unknown initial state

This is a combination of Cases a), b) and c). The model contains no rotation or topography and is initiated from the wrong initial value of ϕ, and in the background solution, no motion is generated. Fig. 7.15 shows the large errors in the background solution, and Fig. 7.16 shows that each of the control vectors significantly reduces this error. However, Fig. 7.14 shows that the solutions produced using the different control vectors are visibly quite different.

When the initial state is used as the control vector, the wrong initial value

of ϕ is corrected, except at the ridge. The omitted topography and rotation are compensated for as when the initial state is used as the control vector in Cases b) and c). After the assimilation, quite large errors remain in the ϕ-field.

When the correction term is used as the control vector, the ϕ- and n-fields are quite close to the true solution at the end of the assimilation interval, but the m-field has larger errors. The solution no longer underestimates the ϕ-field at the end of the assimilation interval as was the case using the correction term as the control vector in Case a). In this case using the correction term happens to give the best fit to the true solution at the end rather than in the middle of the assimilation interval. This could be explained by the fact that the correction to the ϕ-field, shown in Fig. 7.20, is much smaller than it is in Case a) (Fig. 7.17). The correction to the m-field in (Fig. 7.20) is similar to that obtained in Case b) (Fig. 7.18), but is slightly larger. This might explain why there are larger errors in the m-field at the end time in this case than in Case b). The n-field produced is very similar to that produced in Case c) using the correction term as the control vector.

The results using both control vectors together in this case are very good as can be seen by comparing Fig. 7.14 with the true solution of Fig. 7.13. Fig. 7.16 shows that indeed the errors using both control vectors are much smaller than those using either one of the control vectors, and that these errors are almost zero except for those in the n-field. The results shown in the figures were obtained using $q = 1$, and the iteration was terminated after 300 iterations, before the convergence criterion had been satisfied.

Experiments using other control vectors

We mention briefly an attempt at using a couple of the other control vectors mentioned in Chapter 6, Section 6.2 in Experiment 1. We used the spectral form of model error, and a piecewise constant form using three subintervals. Using the spectral form of model error for Cases b) and c), the results were similar to the results obtained using the correction term which we described here. This time, however, more iterations of the descent algorithm were needed. We do not show these results here.

Using the piecewise continuous form, problems arose in the iteration process which were probably due to large differences between the three correction terms. It should be possible to rectify this situation, however, and this would be an interesting topic for further work.

7.4.2 Experiment 2: Fewer observations and observational error

The aim of Experiment 2 is to check whether the conclusions of Experiment 1 still hold in the presence of observational error and when there are fewer observations available. We therefore repeat Experiment 1 Case b) to test these things.

Case e) Observational error

Fig. 7.21 shows the error-corrupted observations used in the assimilation in this case. This noise in the observational data in fact has very little impact on the results, as Fig. 7.22 shows. This indicates that the assimilation effectively filters out the observational noise.

When the initial state is used as the control vector, the initial state found is not completely smooth, but at later times the solution is smooth. The same is true when both the initial state and control vector used as control vectors.

Using the correction term as the control vector in with $q = 1$, the observational error has no visible impact on the results.

Case f) Fewer observations

In this case, observations available every fourth spatial grid point. When the initial state is used as the control vector, the initial state produced by the assimilation is very spiky, although at later times the solution matches the true solution well. This is just as seen in the experiments of Chapters 5 and 6 when fewer observations are available. This highlights again the need to impose extra conditions for smoothness on the initial state found in the assimilation. Using fewer observations also slows the rate of convergence of the iteration process. When the initial state or the correction

term is used as a control vector, approximately three times as many iterations were required. Surprisingly, though, the number of iterations required when both control vectors are used together is about the same as when the whole set of observations are used.

When the correction term is used as the control vector with $q = 0$ (Fig. 7.23), the solution produced by the assimilation is very spiky throughout the assimilation interval, as was found in the experiments of Chapter 5. Also as in Chapter 5, increasing the value of q smoothes the solution, Fig. 7.24 shows the results obtained using $q = 1$. However, in this case the results using $q = 10$ although smooth, were much less accurate. It may be, then, that an alternative method for smoothing the solution is needed when using the correction term as a control vector with fewer observations available.

When both control vectors are used together, the solution obtained is significantly smoother than when either the initial state or the correction term is used alone. Using $q = 0$ produced a smoother initial state than using $q = 1$, but using $q = 1$ produced a smoother solution at later times than using $q = 0$ (Fig. 7.23 and Fig. 7.24).

7.4.3 Experiment 3: The impact of assimilation on a forecast

Experiments 3g and 3h were first carried out performing an assimilation over just 50 timesteps (rather than 100 timesteps as in Experiments 1 and 2), or on the time interval $t \in [0, \frac{T}{2}]$, using the results to initiate a forecast for the interval $[\frac{T}{2}, T]$. The experiments were then repeated using assimilation and forecast intervals of 100 timesteps each. In this case the assimilation was carried out on the interval $[0, T]$, and the results used to initiate a forecast for the interval $[T, 2T]$.

Case g) Omitted topography

Fig. 7.25 shows the true solution over the forecast interval of $t \in [\frac{T}{2}, T]$. It also shows the forecast generated from the true state at time $\frac{T}{2}$ using the imperfect model. This

demonstrates the effects of the model error over this time interval. Starting a forecast with an imperfect model at time $\frac{T}{2}$ from the true state at that time is equivalent to suddenly removing the topography in the middle of a model run. In addition to the existing motion, there are now also waves travelling towards the centre as the fluid fills the area where ridge used to be. Because of this, the forecast from the true state at time $\frac{T}{2}$ using the imperfect model very quickly diverges from the true solution.

We now describe the results of starting a forecast from the assimilation analysis at time $\frac{T}{2}$, comparing the results using the initial state and using the correction term as the control vector. These results are shown in Fig. 7.26.

When the initial state is used as the control vector over the assimilation interval $[0, \frac{T}{2}]$, the ensuing forecast is very similar to the solution obtained by continuing the assimilation over the interval $[\frac{T}{2}, T]$, as comparing Fig. 7.26 with Fig. 7.6 shows. The forecast is a fairly good approximation of the true solution, except at the position of the ridge. In this case the same model is used in the assimilation and in the forecast.

When the correction term is used as the control vector over the assimilation interval $[0, \frac{T}{2}]$, the solution produced at time $\frac{T}{2}$ is in good agreement with the true solution at this time. If this correction term is not included in the ensuing forecast, the impact is similar to that of starting a forecast with an imperfect model from the true solution, which we described above. In the centre of the domain, the forecast very quickly diverges from the true solution. Since the background solution is nearer to the true solution in this region, the assimilation has had a negative impact on the forecast here.

However, if the correction term is included in the forecast, the model used for the forecast is the same as that used in the assimilation. In this case the forecast is much improved and the above problem does not occur. The shape of the solution matches that of the true solution quite well, except for the m-field at the centre of the domain. The forecast over the interval $[\frac{T}{2}, T]$ is on the whole better than the forecast obtained from the assimilation using the initial state as the control vector.

We now describe the results of performing the assimilation and the forecast over longer time intervals. Fig. 7.27 shows the true solution on the time interval $[T, 2T]$.

Over this time interval, the waves reach the boundaries of the domain and start to travel back towards the centre. Fig. 7.27 also shows the forecast obtained with the imperfect model started from the true model state at time T. Since this forecast is over a period twice as long as before, the forecast diverges even further from the true model state in this case.

Fig. 7.28 shows the forecasts ensuing from assimilation intervals using the different control vectors. If the initial state is used as the control vector for assimilation over the interval $[0, T]$, the ensuing forecast still gives a significant improvement over the background solution for the same period in the m- and n-fields, but the forecast of the ϕ-field hardly improves on the background solution.

When the correction term is used as the control vector in the assimilation over the interval $[0, T]$, the solution produced is a very good approximation of the true solution, and so the forecast not including the correction term quickly diverges from the true solution. However, when the correction term is included in the forecast, the forecast is quite close to the true solution. It seems unlikely that the correction term found over the assimilation interval $[0, T]$ really compensates for model error in the forecast interval $[T, 2T]$, since the motion in each interval is in opposite directions. It is more likely that the forecast is good because it is started from an estimate close to the true state, and it does not diverge quickly from the true solution because there is no difference in the model used for the forecast and in the assimilation.

Case h) Omitted rotation

Fig. 7.29 shows the true solution on the time interval $[\frac{T}{2}, T]$, and also the forecast with the imperfect model initiated from the true solution at time $\frac{T}{2}$, which shows the effects of model error over the forecast interval.

When a forecast with the imperfect model is performed starting from the true solution at time $\frac{T}{2}$, it is as if the Coriolis parameter f is suddenly set at zero in the middle of a model run. However, the impact of this is very gradual and only affects the n-field. Hence, the forecast using the imperfect model started from a true state in this case diverges only very slowly from the true solution.

Fig. 7.30 shows the forecasts ensuing from assimilation using each of the different

control vectors. If the initial state is used as the control vector in the assimilation interval, the ensuing forecast shows an improvement over the background solution in the middle of the forecast, but not at the end; all the benefit of the assimilation is lost by the end of the forecast.

However, when the correction term is used as the control vector during the assimilation interval, the solution at time $\frac{T}{2}$ is closer to the true state, and hence the forecast is better than when the initial state is used as the control vector in the assimilation. This is true whether the correction term is included in the forecast or not, and it is hard to judge whether or not including it is beneficial in this case.

Experiment 3h was repeated using the longer assimilation and forecast intervals. The results for this case are shown in Fig. 7.31 and Fig. 7.32. Here, much the same conclusions hold as for the shorter time interval, except that in this case a better forecast is achieved by using the correction term in the assimilation but not in the forecast.

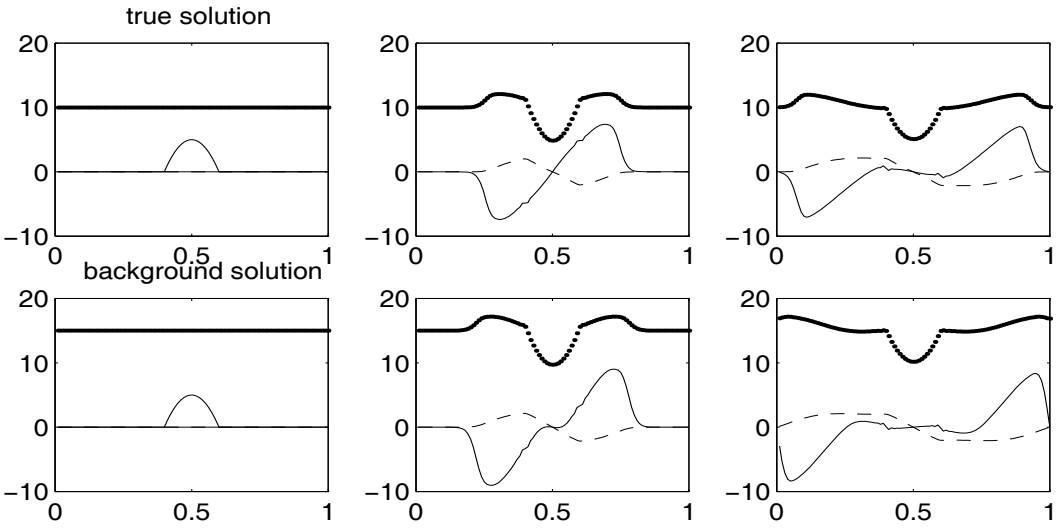

Figure 7.1: Experiment 1a: perfect model, wrong initial state. The true solution and background solution (no assimilation) at the beginning, middle and end of the assimilation interval. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

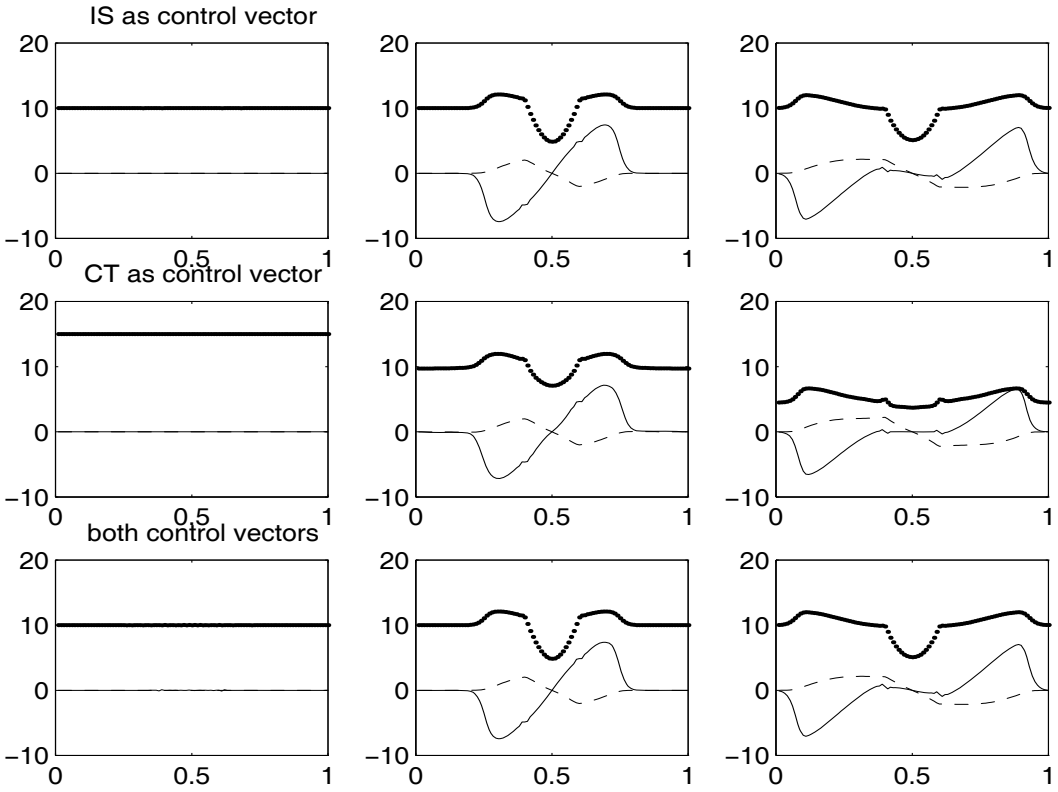

Figure 7.2: Experiment 1a: perfect model, wrong initial state. The solutions after assimilation using the initial state (IS), the correction term (CT) and both together as control vectors. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

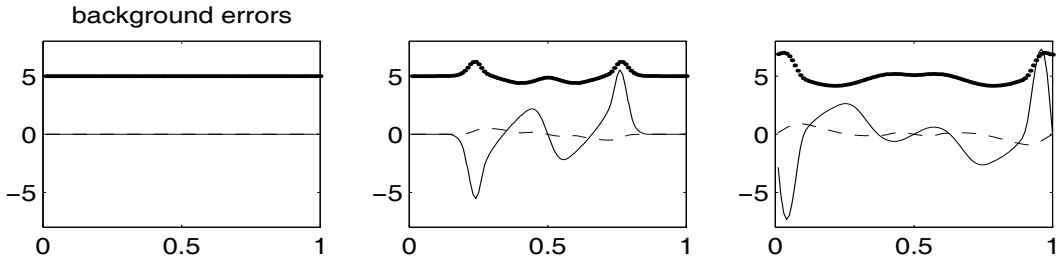

Figure 7.3: Experiment 1a: perfect model, wrong initial state. The errors in the background solution (errors before assimilation). Dotted line: errors in the ϕ-field; dashed line: errors in the n-field; solid line: errors in the m-field

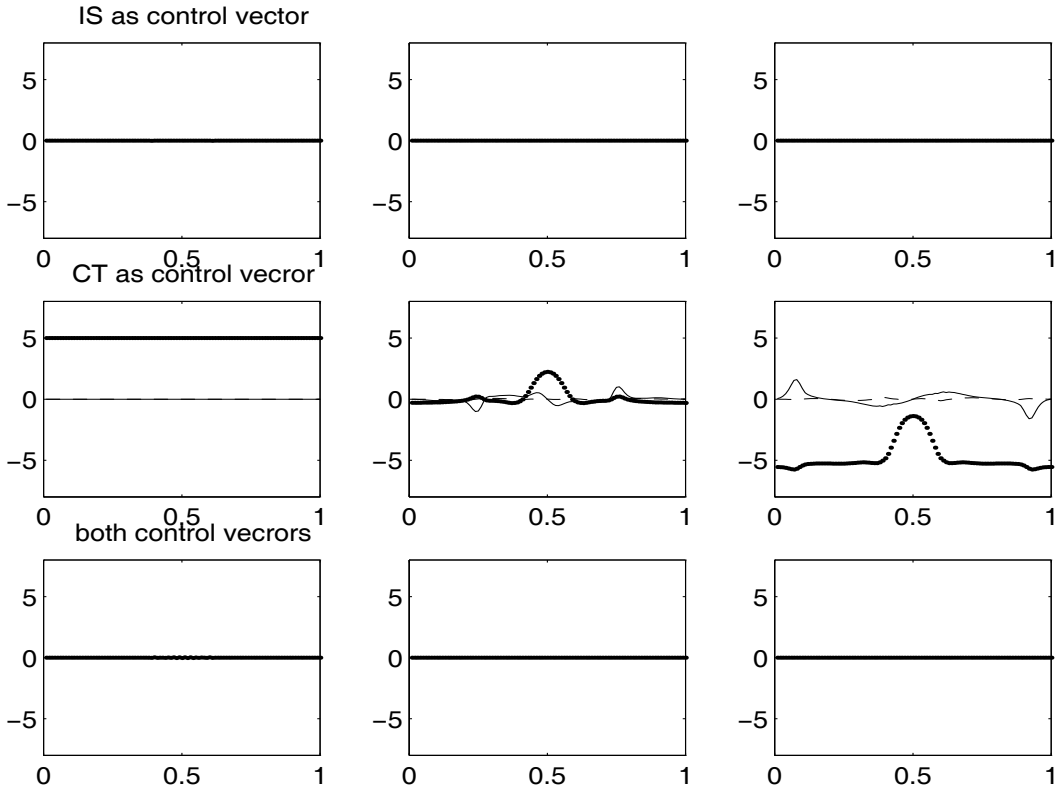

Figure 7.4: Experiment 1a: perfect model, wrong initial state. The errors in the solutions after assimilation using the initial state (IS), correction term (CT) and both together as control vectors. Dotted line: errors in the ϕ-field; dashed line: errors in the n-field; solid line: errors in the m-field

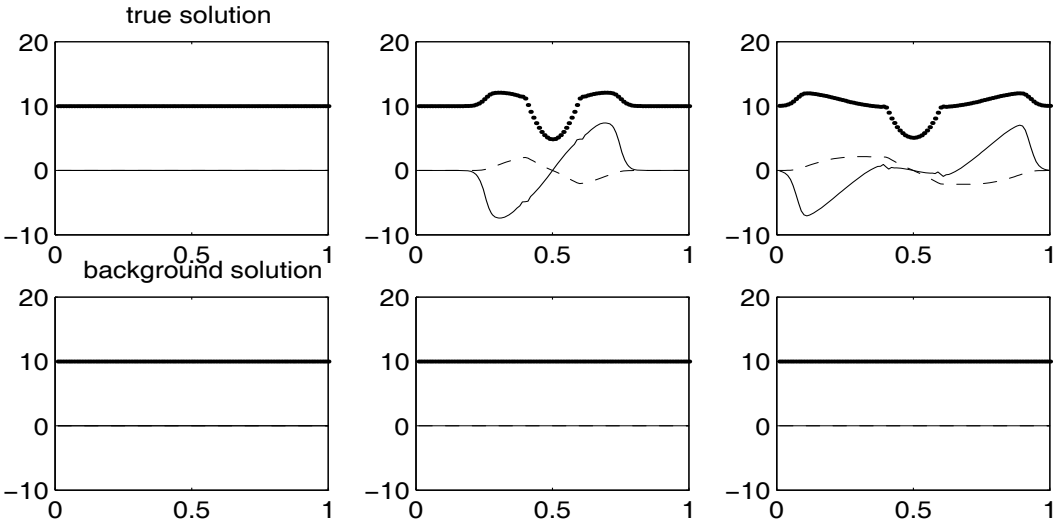

Figure 7.5: Experiment 1b: model error is due to omitted topography. The true solution and background solution (no assimilation) at the beginning, middle and end of the assimilation interval. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

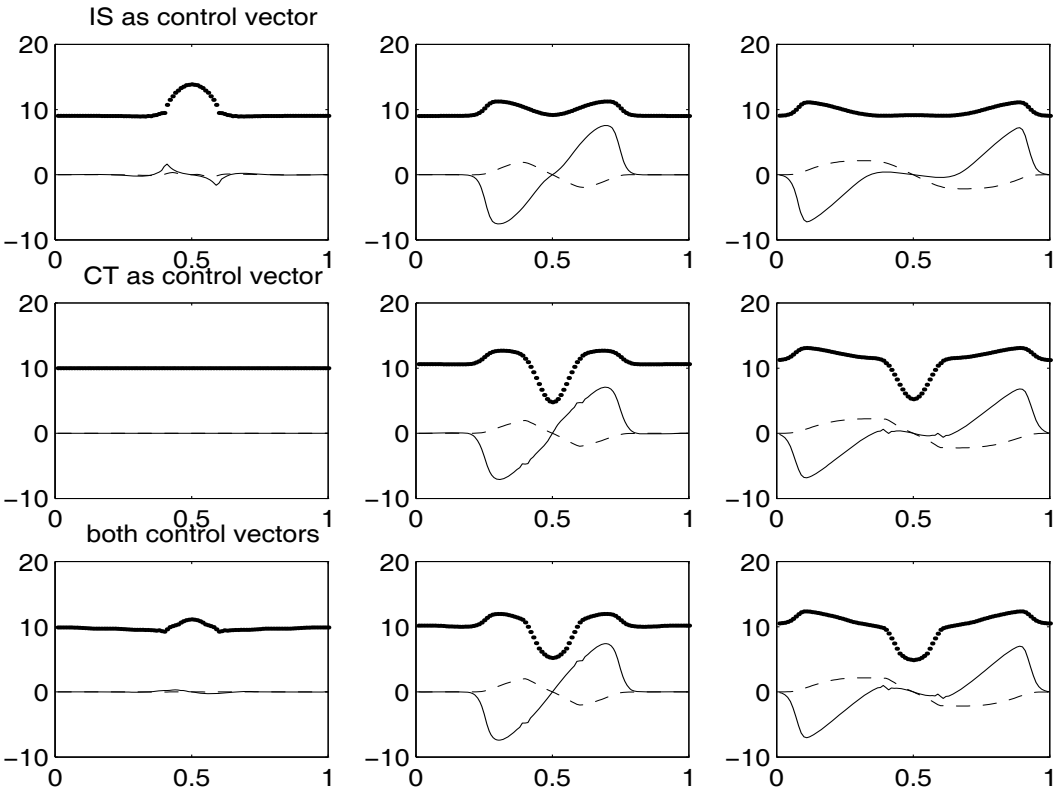

Figure 7.6: Experiment 1b. The solutions after assimilation using the initial state (IS), the correction term (CT) and both together as control vectors.

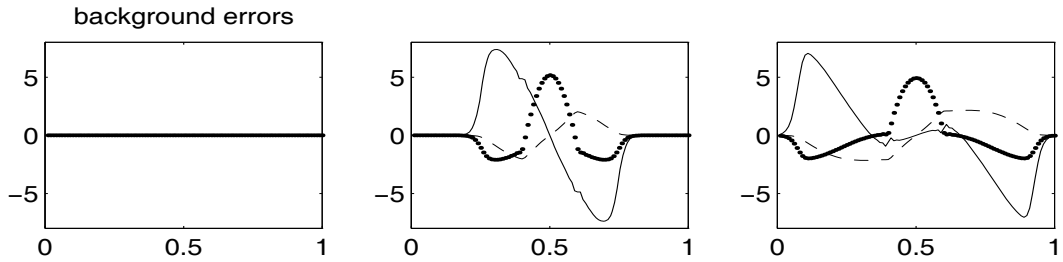

Figure 7.7: Experiment 1b: model error is due to omitted topography. The errors in the background solution (errors before assimilation). Dotted line: errors in the ϕ-field; dashed line: errors in the n-field; solid line: errors in the m-field.

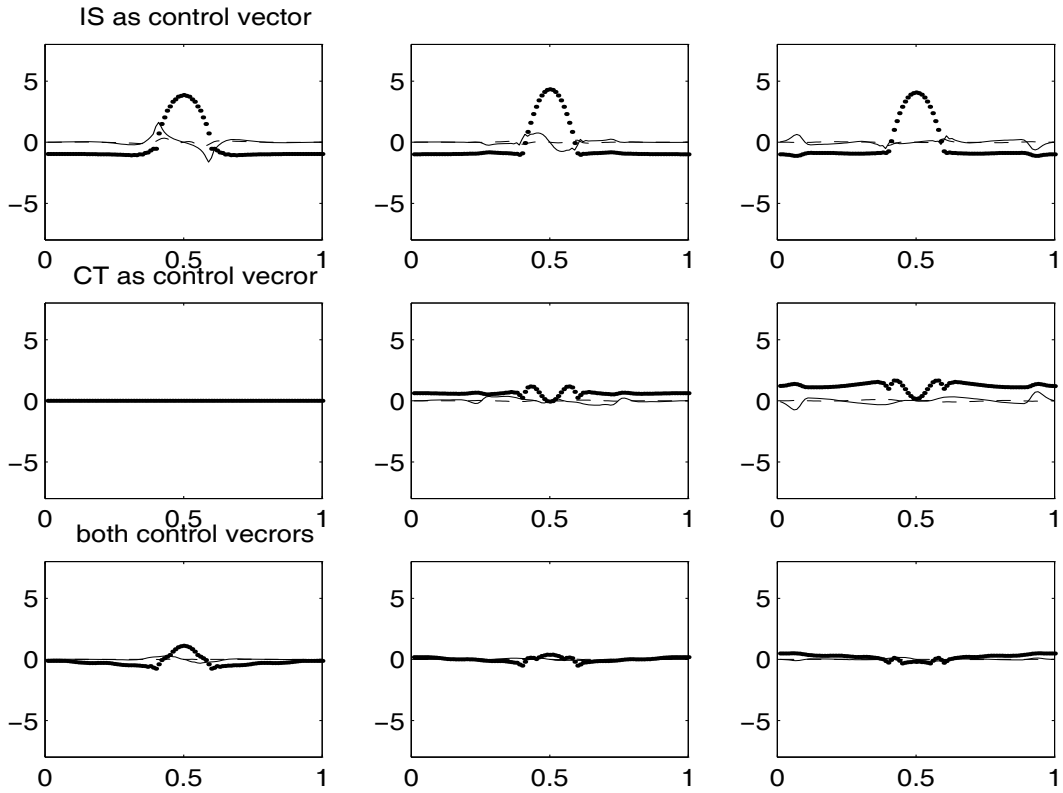

Figure 7.8: Experiment 1b: model error is due to omitted topography. The errors in the solutions after assimilation using the initial state (IS), correction term (CT) and both together as control vectors.

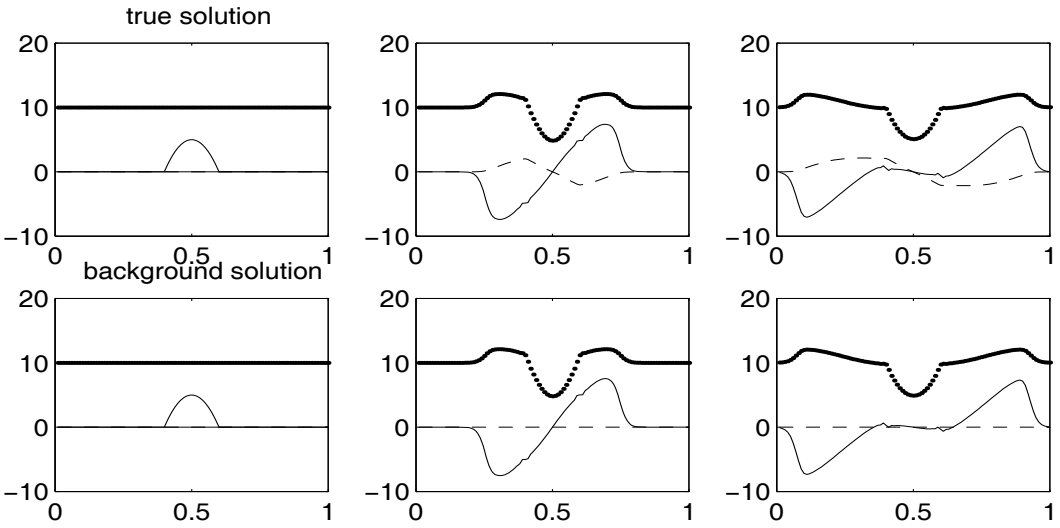

Figure 7.9: Experiment 1c: model error is due to omitted rotation. The true solution and background solution (no assimilation) at the beginning, middle and end of the assimilation interval. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

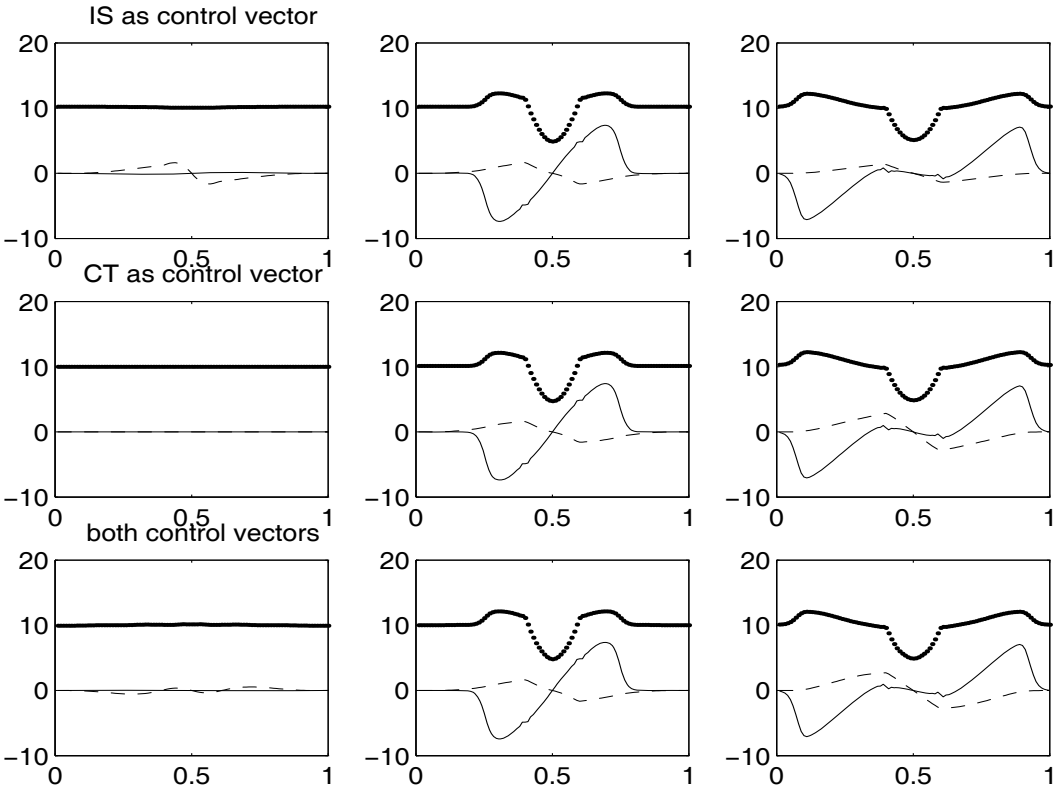

Figure 7.10: Experiment 1c: model error is due to omitted rotation. The solutions after assimilation using the initial state (IS), the correction term (CT) and both together as control vectors.

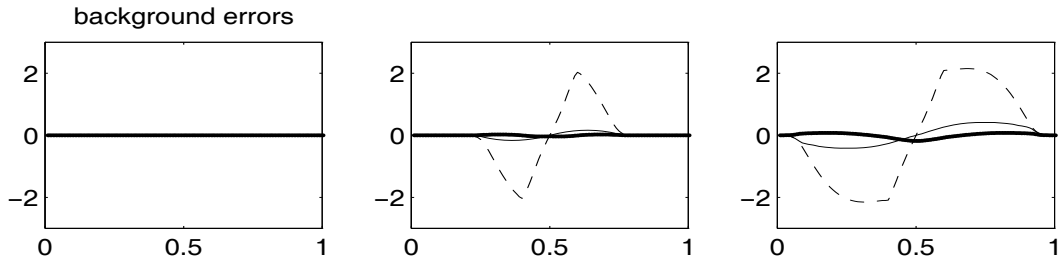

Figure 7.11: Experiment 1c: model error is due to omitted rotation. The errors in the background solution (errors before assimilation). Dotted line: errors in the ϕ-field; dashed line: errors in the n-field; solid line: errors in the m-field.

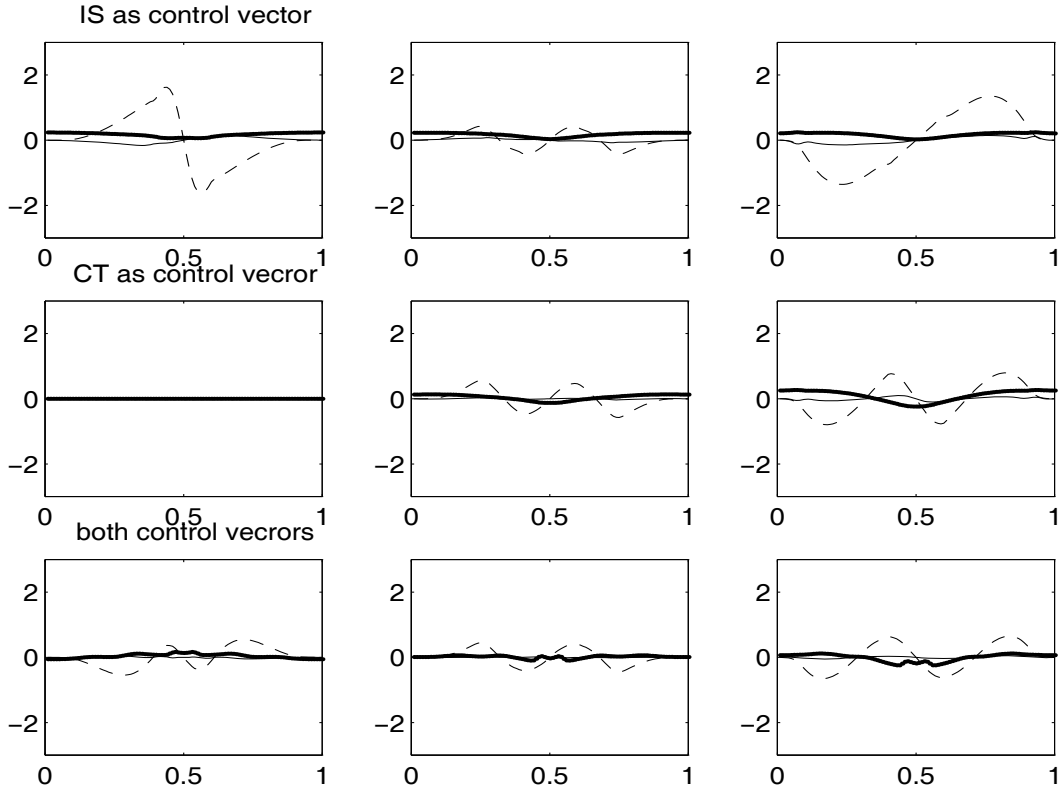

Figure 7.12: Experiment 1c: model error is due to omitted rotation. The errors in the solutions after assimilation using the initial state (IS), correction term (CT) and both together as control vectors.

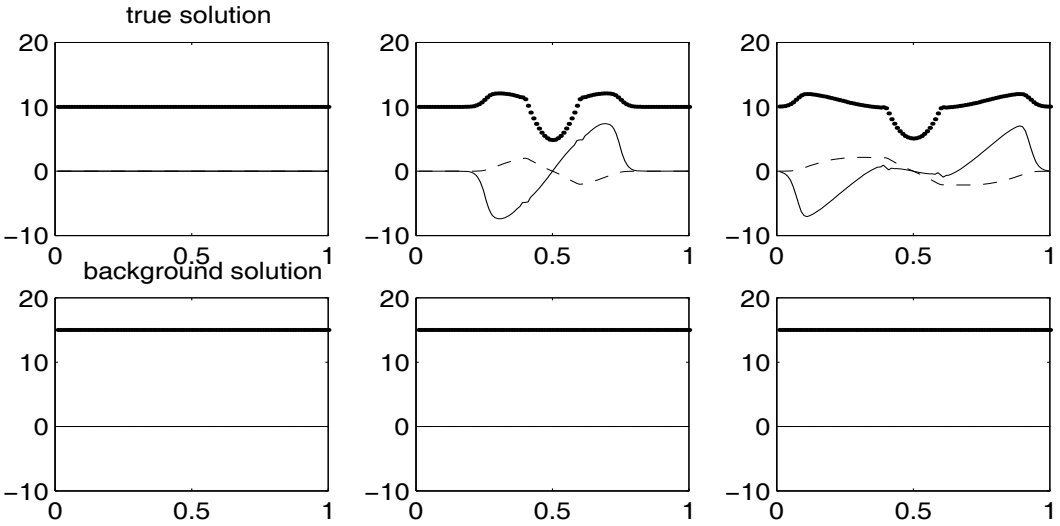

Figure 7.13: Experiment 1d: unknown initial state, and model error is due to omitted topography and rotation. The true solution and background solution (no assimilation) at the beginning, middle and end of the assimilation interval. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

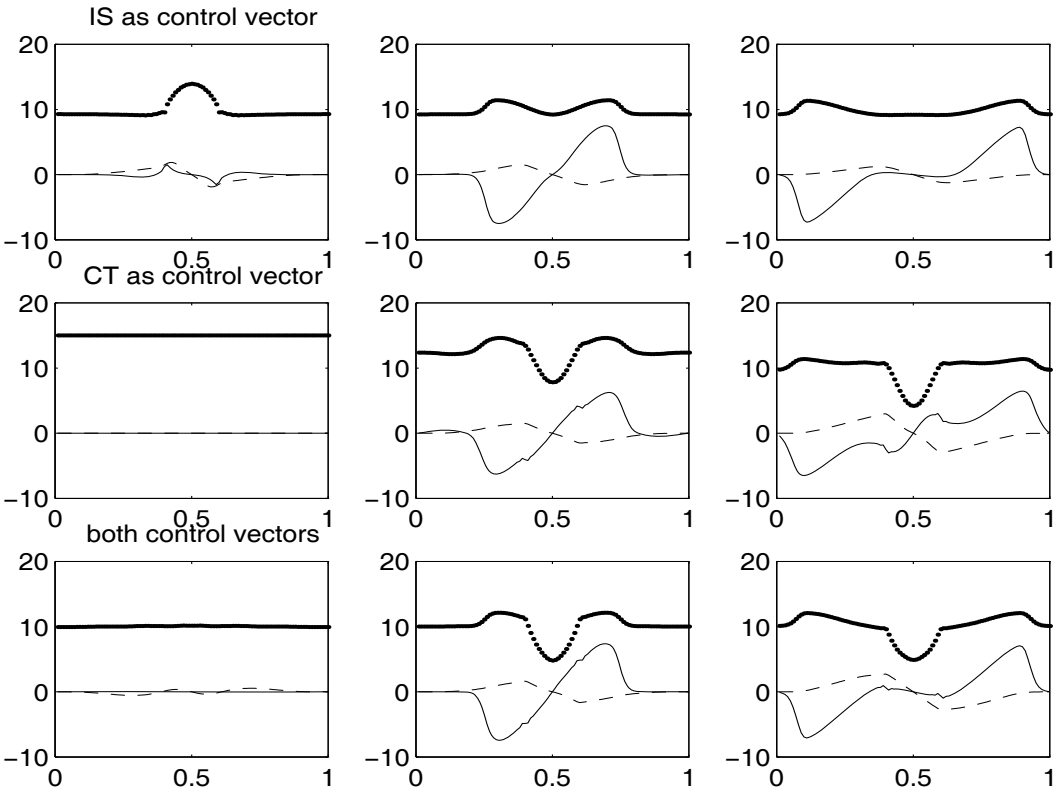

Figure 7.14: Experiment 1d. The solutions after assimilation using the initial state (IS), the correction term (CT) and both together as control vectors.

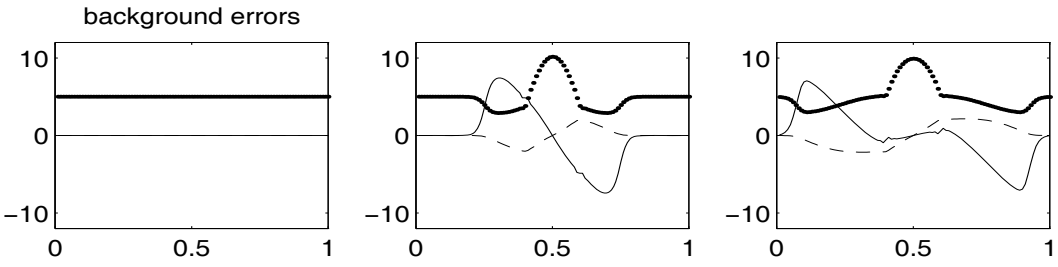

Figure 7.15: Experiment 1d: unknown initial state, and model error is due to omitted topography and rotation. The errors in the background solution (errors before assimilation). Dotted line: errors in the ϕ-field; dashed line: errors in the n-field; solid line: errors in the m-field.

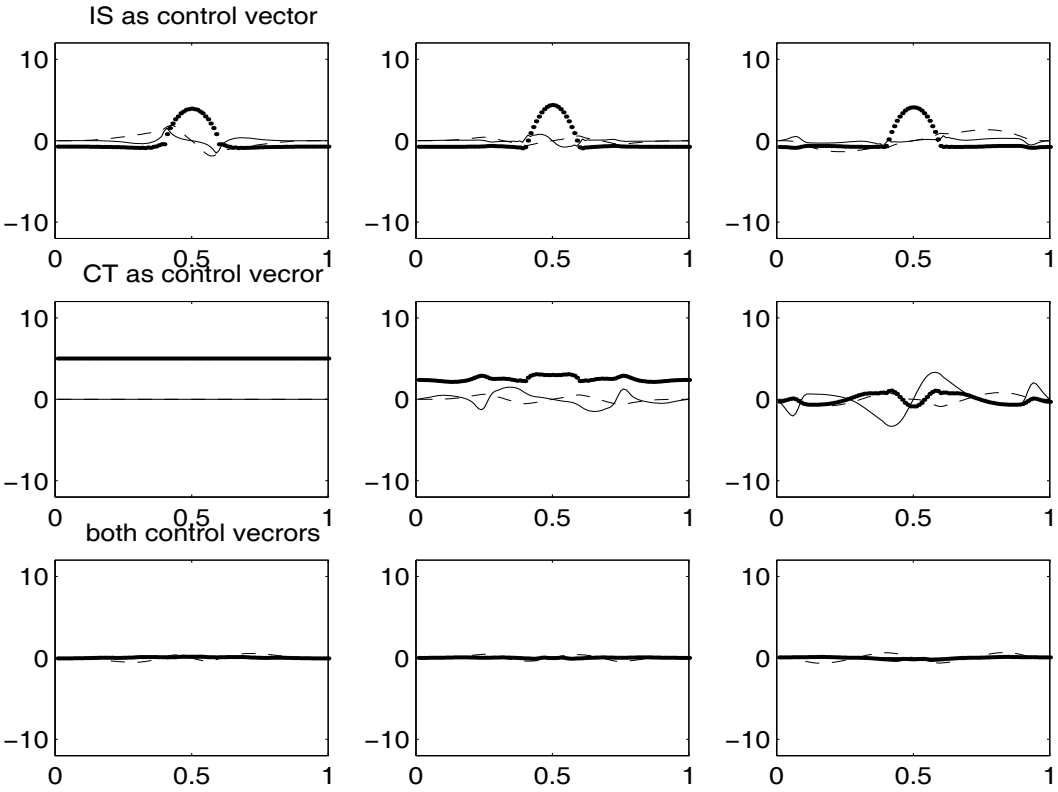

Figure 7.16: Experiment 1d: unknown initial state, and model error is due to omitted topography and rotation. The errors in the solutions after assimilation using the initial state (IS), correction term (CT) and both together as control vectors.

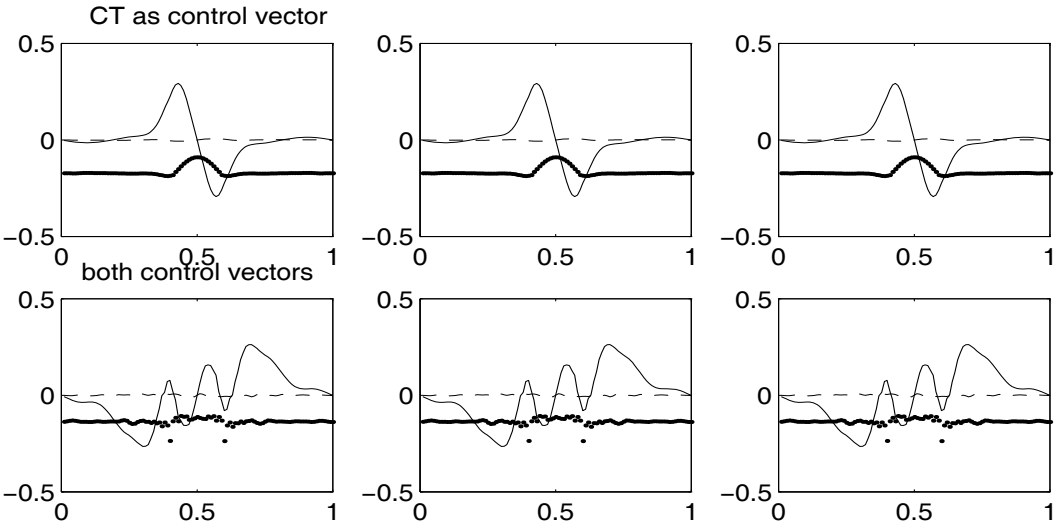

Figure 7.17: Experiment 1a: the correction terms found in the assimilation using the correction term and both the correction term and initial state as control vectors. Dotted line: correction to the ϕ-field; dashed line: correction to the n-field; solid line: correction to the m-field.

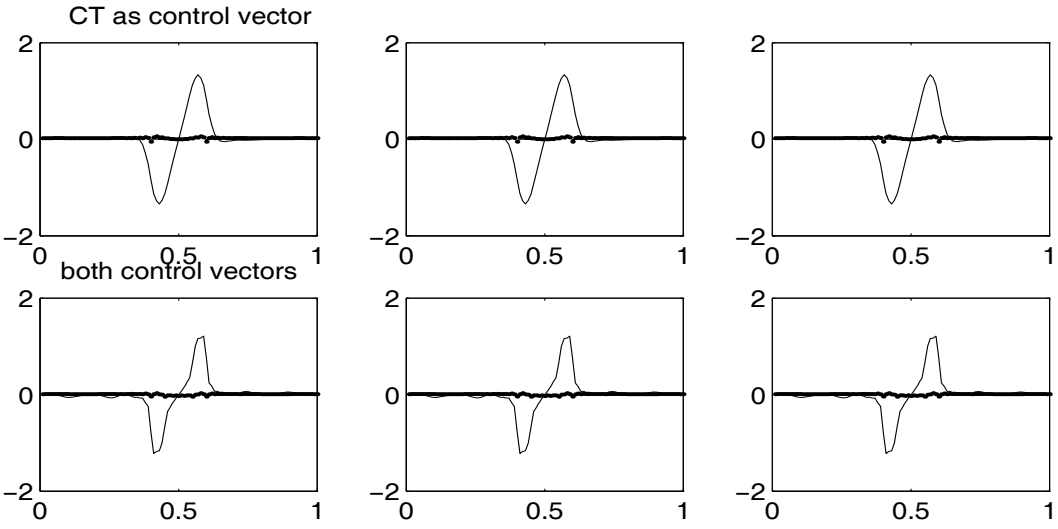

Figure 7.18: Experiment 1b: the correction terms found in the assimilation using the correction term and both the correction term and initial state as control vectors. Dotted line: correction to the ϕ-field; dashed line: correction to the n-field; solid line: correction to the m-field.

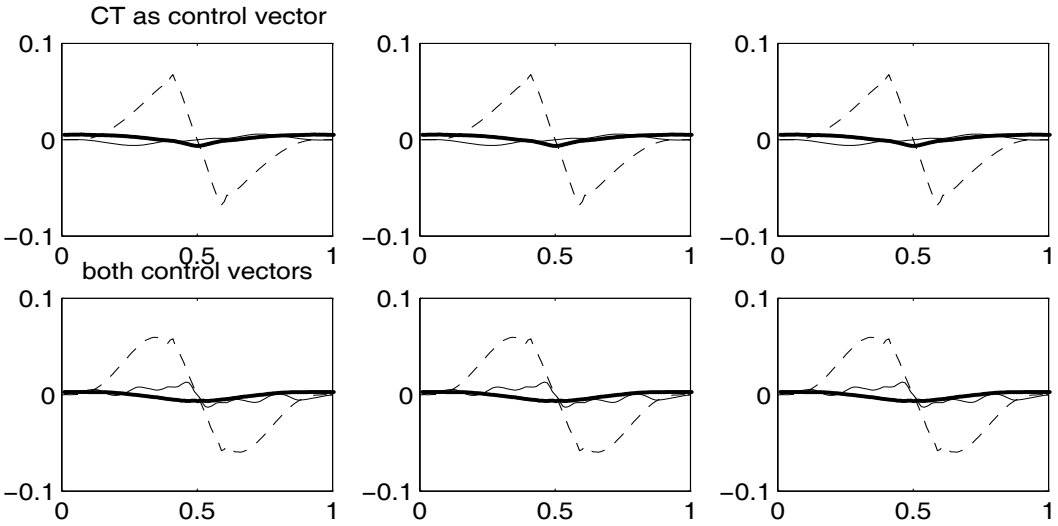

Figure 7.19: Experiment 1c: the correction terms found in the assimilation using the correction term and both the correction term and initial state as control vectors. Dotted line: correction to the ϕ-field; dashed line: correction to the n-field; solid line: correction to the m-field.

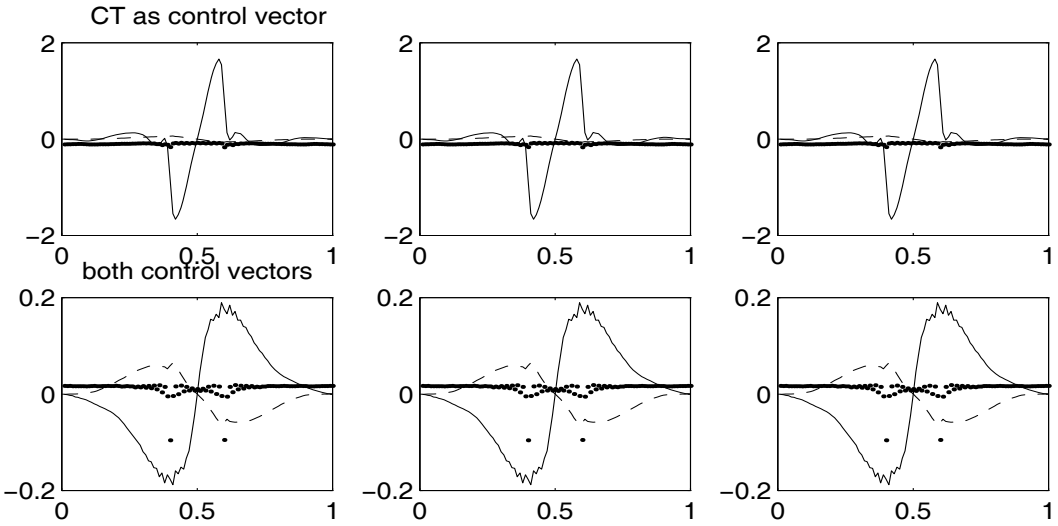

Figure 7.20: Experiment 1d: the correction terms found in the assimilation using the correction term and both the correction term and initial state as control vectors. Dotted line: correction to the ϕ-field; dashed line: correction to the n-field; solid line: correction to the m-field.

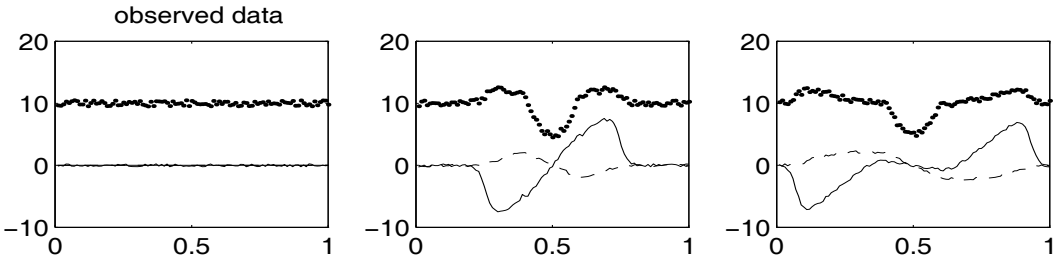

Figure 7.21: Experiment 2e: observations corrupted by random error. The observational data used in the experiments. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

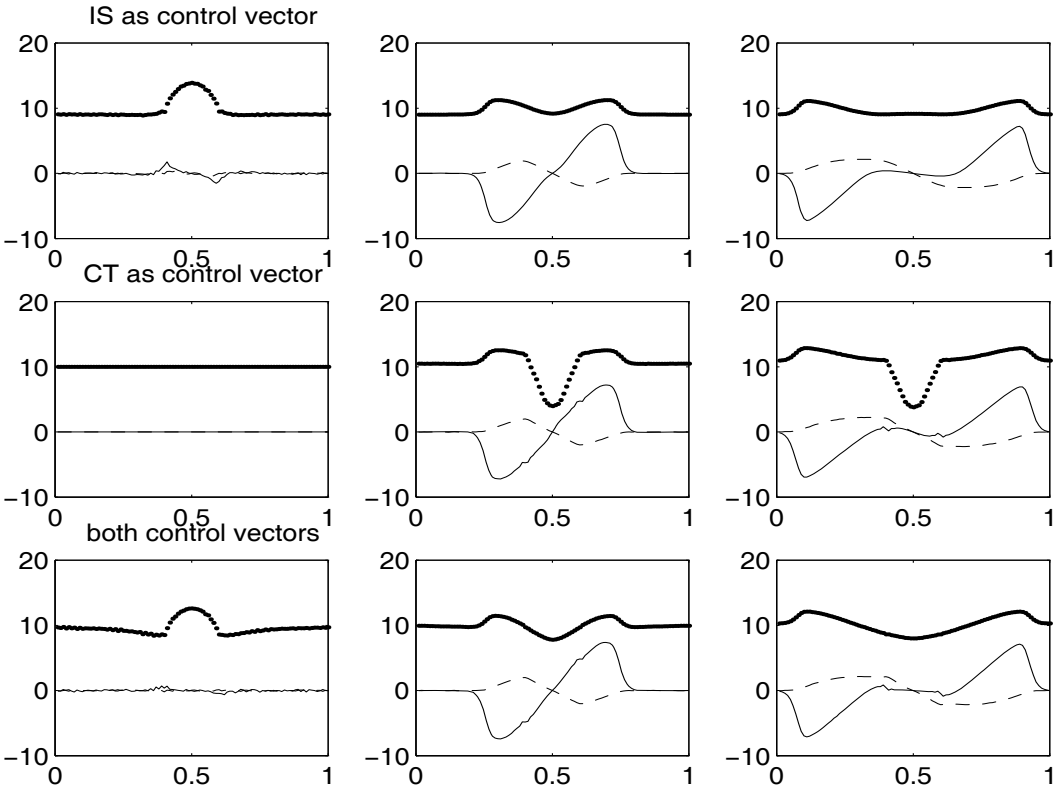

Figure 7.22: Experiment 2e: solutions after assimilation using observations containing error. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

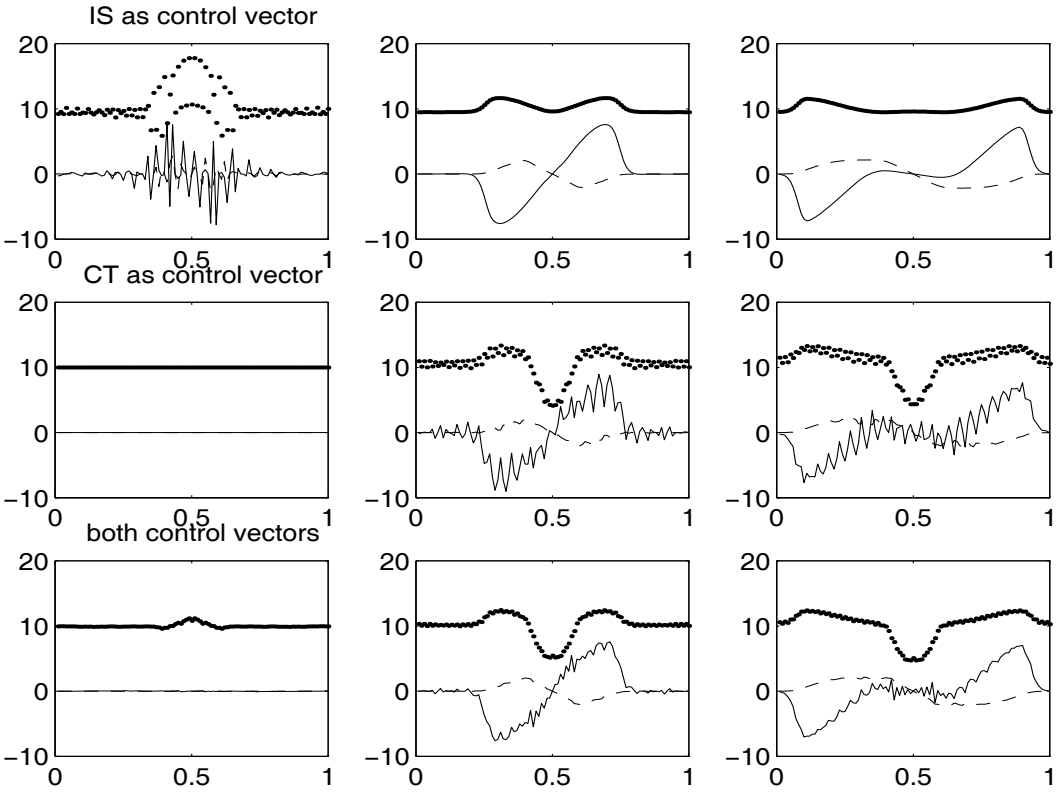

Figure 7.23: Experiment 2f: solutions after assimilation when fewer observations are available, with $q = 0$. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

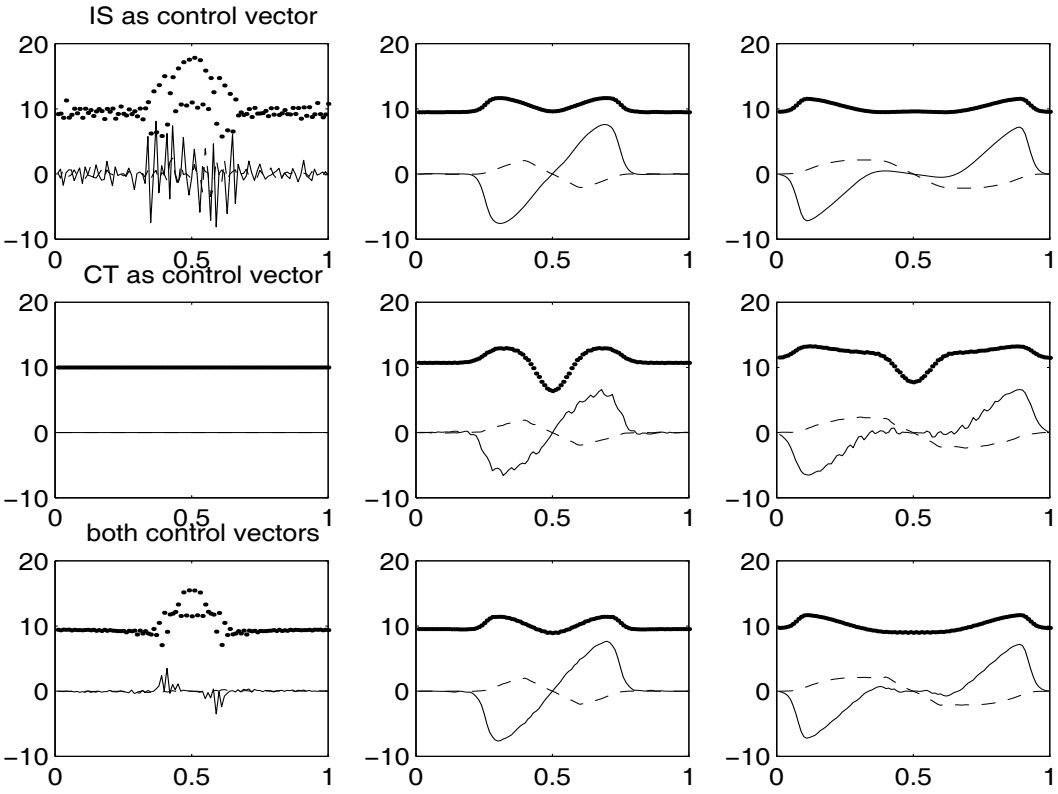

Figure 7.24: Experiment 2f: solutions after assimilation when fewer observations are available, with $q = 1$. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

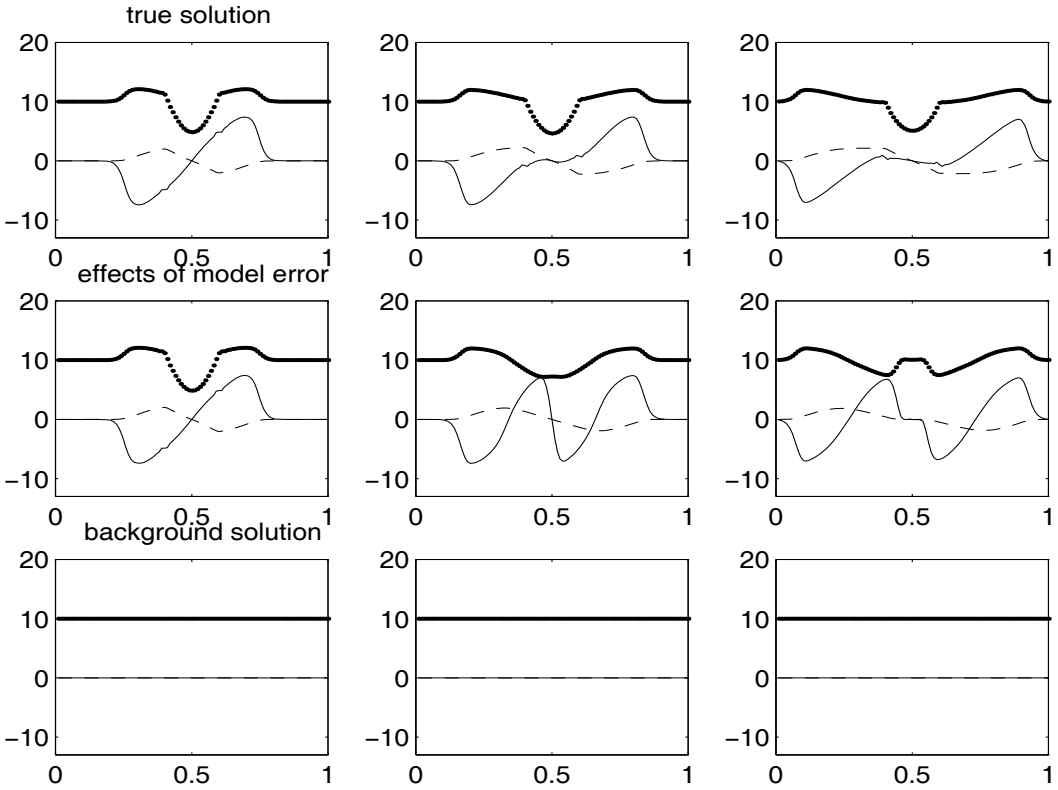

Figure 7.25: Experiment 3g: model error is due to omitted topography; forecast over the interval $t \in [\frac{T}{2}, T]$. The figure shows the true solution, a forecast with the imperfect model started from the true state at time $\frac{T}{2}$, and the background solution. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

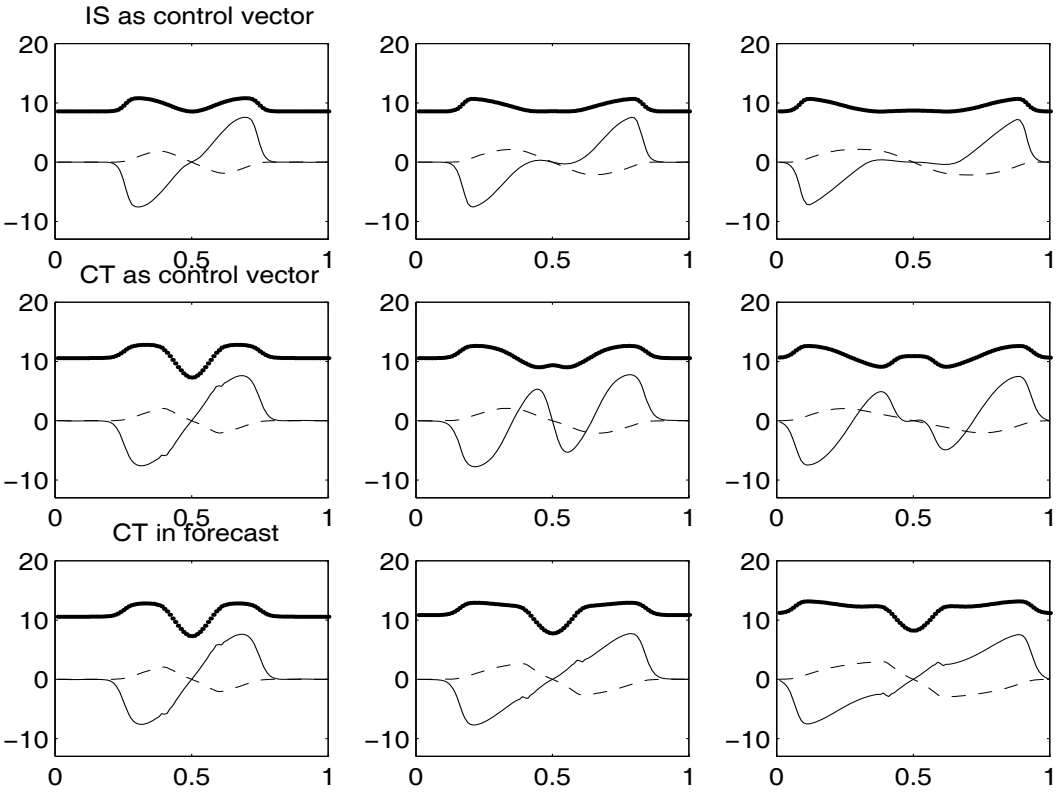

Figure 7.26: Experiment 3g: the effect of assimilation on the interval $t \in [0, \frac{T}{2}]$ using the initial state and using the correction term on a forecast over the interval $t \in [\frac{T}{2}, T]$. In the third row, the correction term is included in the forecast, but in the second row it is not.

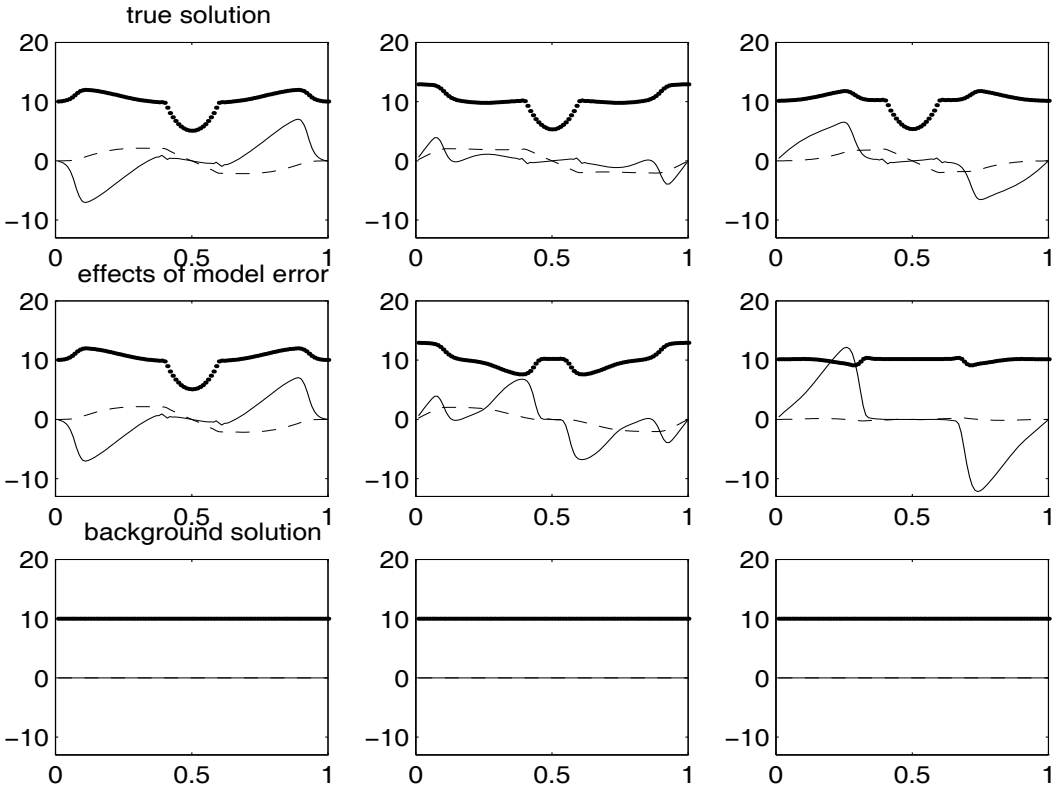

Figure 7.27: Experiment 3g: model error is due to omitted topography; forecast over the interval $t \in [T, 2T]$. The figure shows the true solution, a forecast with the imperfect model started from the true state at time T, and the background solution. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

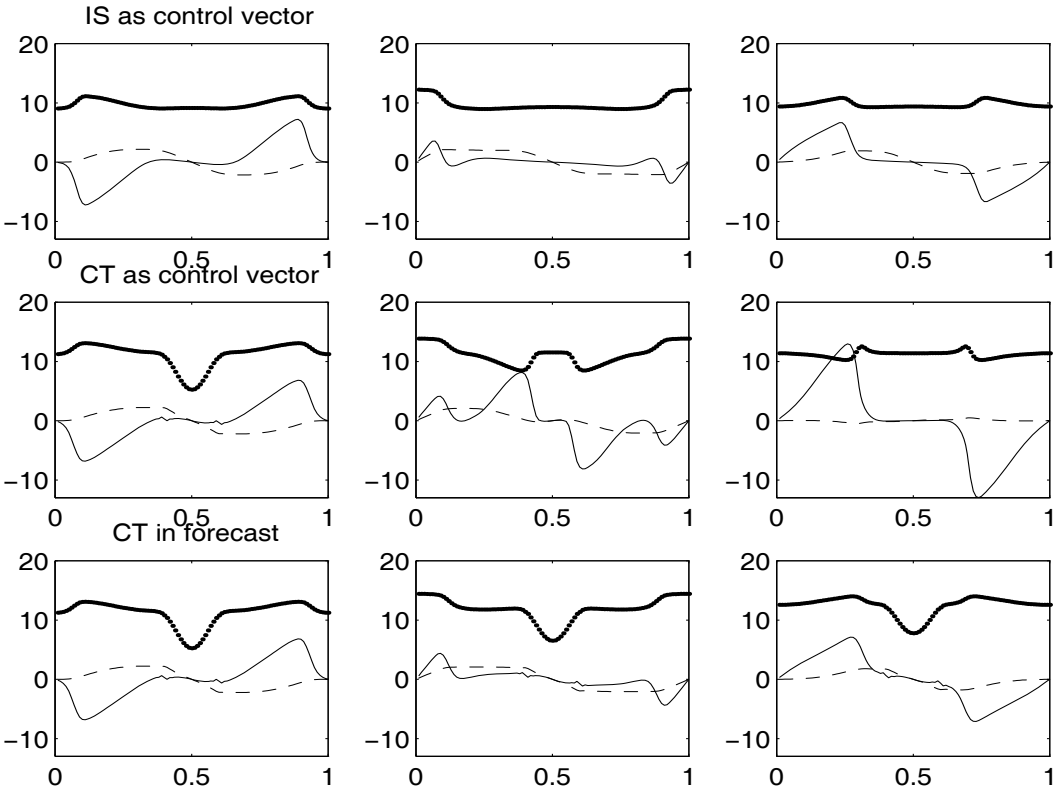

Figure 7.28: Experiment 3g: the effect of assimilation on the interval $t \in [0, T]$ using the initial state and using the correction term on a forecast over the interval $t \in [T, 2T]$. In the third row, the correction term is included in the forecast, but in the second row it is not.

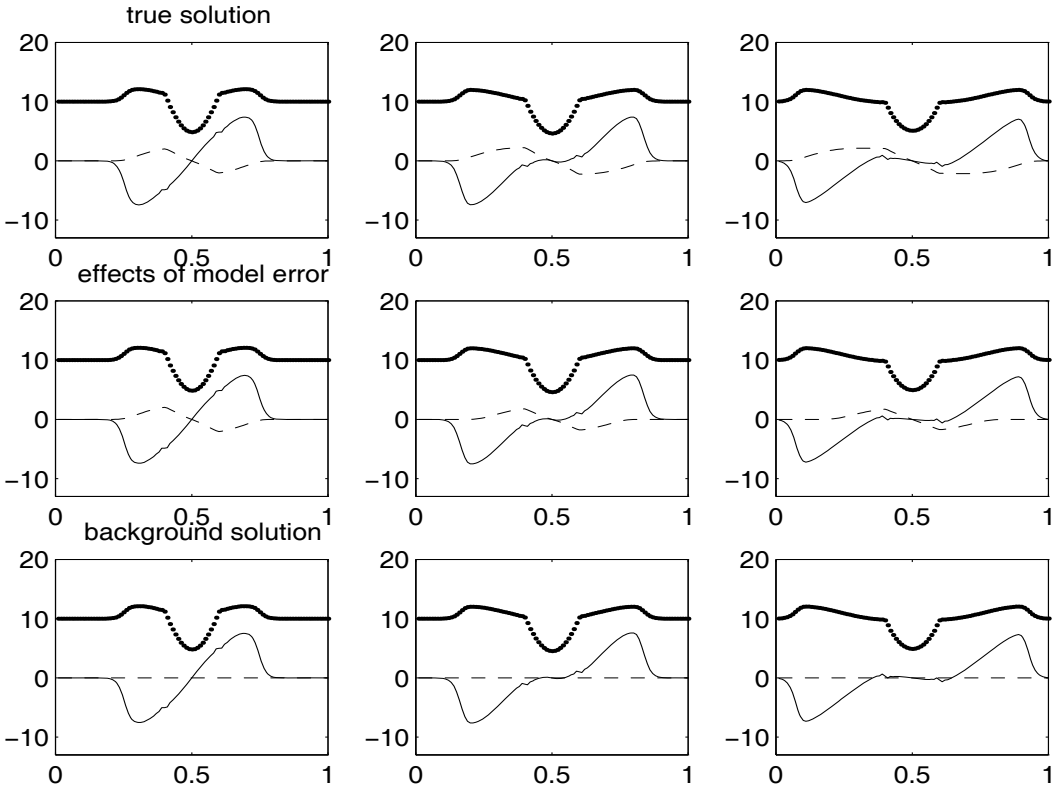

Figure 7.29: Experiment 3h: model error is due to omitted rotation; forecast over the interval $t \in [\frac{T}{2}, T]$. The figure shows the true solution, a forecast with the imperfect model started from the true state at time $\frac{T}{2}$, and the background solution. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

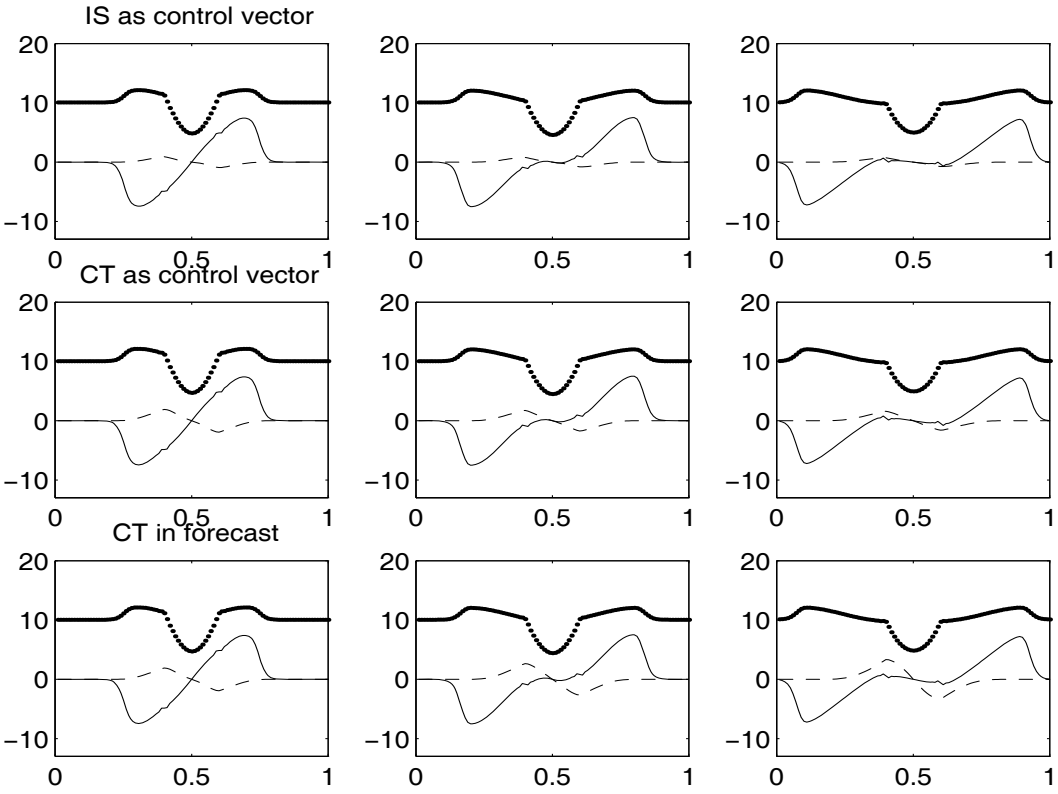

Figure 7.30: Experiment 3h: the effect of assimilation on the interval $t \in [0, \frac{T}{2}]$ using the initial state and using the correction term on a forecast over the interval $t \in [\frac{T}{2}, T]$. In the third row, the correction term is included in the forecast, but in the second row it is not.

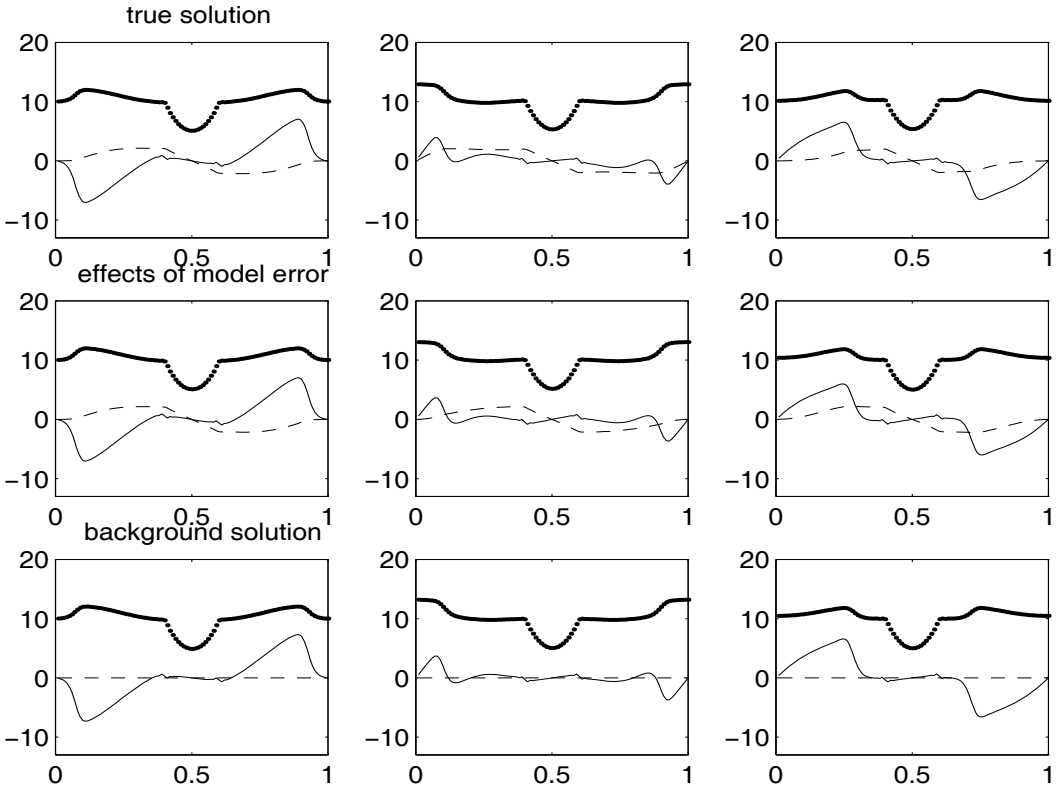

Figure 7.31: Experiment 3h: model error is due to omitted rotation; forecast over the interval $t \in [T, 2T]$. The figure shows the true solution, a forecast with the imperfect model started from the true state at time T, and the background solution. Dotted line: ϕ-field; dashed line: n-field; solid line: m-field.

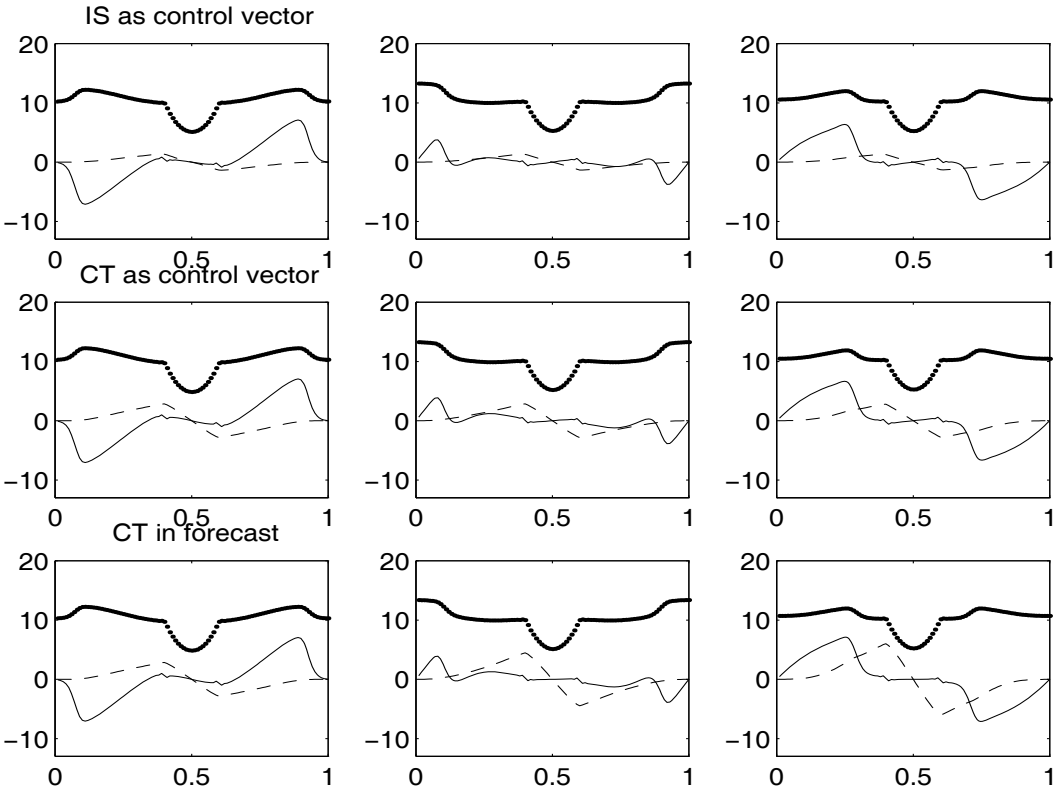

Figure 7.32: Experiment 3h: the effect of assimilation on the interval $t \in [0, T]$ using the initial state and using the correction term on a forecast over the interval $t \in [T, 2T]$. In the third row, the correction term is included in the forecast, but in the second row it is not.

7.5 Summary and Conclusions

Experiment 1: Comparing different control vectors

In Experiment 1 we compared the effectiveness of the initial state, the (constant) correction term and both together as control vectors in compenstating for errors in the initial state, for two types of model error, and for a combination of these errors during an assimilation interval. This extends the experiments of Chapter 5 since the model dynamics are more complex, and because the model error depends on the model state and so is not constant in time. Generally, the conclusions of the experiments of Chapter 5 are found to hold here, too. We found that the initial state can compensate to some extent for model error, generally producing a solution which is closest to the true solution in the middle of the assimilation interval. Similarly, using the constant correction term as the control vector compensates to some extent for the effects of errors in the initial state, and a suitable choice of q can give a solution close to the true solution at the end rather than in the middle of the assimilation interval.

Using the constant correction term as the control vector compensates well for both types of model error investigated, and reduces the errors in the background solution (with no assimilation) more than using the initial state as a control vector. Since the model error in each case depends on the model state, and since the motion during the assimilation interval propagates almost half way across the model domain in each direction, it is significant that the constant correction term can compensate for model error on this time scale. In each case the correction term found in the assimilation seems to represent a temporal average of the actual model error. When using a correction term in the assimilation, we are altering the model equations over the assimilation interval. It is therefore important to check that what the correction term represents makes sense, as Wergen argues [87]. Here, it seems that the correction term does in fact approximate model error.

Using both control vectors together requires many more iterations but is particularly successful in Case d) in which there are errors in initial state as well as two sources of model error. These results show that using both control vectors together

could be very effective, but in our experiments the number of iterations required is unacceptably high. To alleviate this situation, ways of effectively preconditioning the problem should be investigated.

The results seem to depend quite strongly on the choice of q, or on how strongly (if at all) the correction term is constrained to be small. As in Chapter 5, however, it was found that if the correction term is being used to compensate for errors in the initial state, then a larger value of q should be used. If on the other hand the correction term is compensating for model error, using a small value of q gives more accurate results, but larger values of q requires fewer iterations of the descent algorithm. In our examples, using $q = 1$ gives a good compromise between accurate solutions and a reasonable number of iterations. These conclusions on the choice of q are similar to those made in the experiments of Chapters 5 and 6.

Experiment 2: Fewer observations and observational errors

The results from Experiment 2 show that whereas random observational errors do not have a big impact on the assimilation, using fewer observations does. The results were much the same as the results from the experiments of Chapter 5 and of Chapter 6 using the initial state or the constant correction term as a control vector when a quarter of the observations are available. When the initial state is used as the control vector, the initial state produced in the assimilation is not smooth, although this has little impact on the solution at later times. This highlights the need to impose extra conditions for smoothness of the solution when a full set of observations is not available, for example by constraining the initial state to be close to a background value. A background term was not included in our experiments because our primary aim was to compare the control vectors in the idealised case of a full set of observations.

When the correction term is used as the control vector, increasing the value of q (and hence constraining the correction term to be small) helps to smooth the solution. However, since increasing the value of q leads to less accurate results, an alternative way of ensuring smoothness might prove more successful.

Experiment 3: The impact of assimilation on the forecast

In Experiment 3 we test whether the improvement in the solution at the end of an assimilation interval, produced by assimilation using each of control vectors, results in an improvement in a subsequent forecast. This is carried out for both examples of model error. In the case that model error is due to the omission of topography, using the initial state as a control vector gives a significant improvement over the background solution not only in the assimilation interval, but also in the forecast.

Using the constant correction term as a control vector gives a good improvement during the assimilation interval, but if the correction term is not included in the forecast, the forecast soon deteriorates, and becomes worse than if no assimilation had been performed. These results can be explained by examining the impact of starting a forecast with an imperfect model from the true solution at the end of the assimilation interval. It is the impact of using a different model for the assimilation and forecast that causes the forecast to deteriorate quickly. We note that this situation could be alleviated by gradually phasing out the correction term during the assimilation interval using the predetermined scalars of equation (4.32), as was done in the original paper on the correction term technique [26]. When we include the correction term in the forecast, however, the forecast is good; better than the forecast produced using the initial state as the control vector in the assimliation.

When the model error is due to omission of rotation, slightly different conclusions are reached. In this case, using the initial state as the control vector in the assimilation results in a slightly improved forecast, but the benefits of the assimilation are lost by the end of the forecast. When the correction term is used in the assimilation, the impact of including the correction term in the forecast dependes on the length of the assimilation and forecast intervals. For the shorter intervals, there is a slight improvement to the forecast whether or not the correction term is included in the forecast. For the longer time intervals, however, better results are obtained by not including the correction term in the forecast.

From these results, we see that some attention is needed on the issue of whether or not to use the correction term in the forecast. The results from the experiments with omitted topography indicate the danger of suddenly cutting out the correction

term. The results from the experiments with omitted rotation, on the other hand, indicate the danger of leaving the correction term in the forecast for too long. It seems that on the whole, it would be best to gradually phase out the correction term during the forecast interval. It would be worth carrying out further work to investigate this.

These results show, however, that using a correction term in the assimilation to compensate for model error, it is possible to produce a better forecast than can be produced using the initial state in the assimilation. In these experiments the model error depends on the model state, and the assimilation and forecast intervals represent a significantly long timescale.

Chapter 8

Conclusions

In this thesis, we have considered problems in data assimilation using a framework of control theory. In Chapter 3, we showed how a successive correction method of data assimilation could be regarded as an observer if observations are available frequently in time. Observer theory can be used to design the weighting matrices so that the data assimilation scheme has desirable dynamical properties. In particular, we gave conditions under which the analysis converges in time to the true solution, for the linear time-invariant case. In our experiments using a simple model, an observer designed for good temporal convergence gave much faster convergence than the Cressman successive correction scheme in areas distant from the observation positions. These results serve to illustrate the potential benefits of suitable observer design.

The majority of the work in the thesis is geared towards the 4D variational approach to data assimilation. In particular, we address the problem of how to account for model error without incurring unreasonable extra expense. One method for doing this is the correction term technique [26], in which a constant correction term approximating model error is added to the model equations. The correction term is then used instead of, or as well as, the model initial state as a control vector in variational assimilation.

In the context of a linear, time varying system, we looked for conditions for uniqueness of solutions of the 4D variational assimilation problem using the initial state, the correction term or both together as control vectors. When the initial state

is used as the control vector, complete N-step observability at time t_0 is a necessary and sufficient condition for uniqueness in the case where the cost function consists of observations from the time interval $[t_0, t_{N-1}]$. We showed however, that complete N-step observability at time t_0 is neither a necessary nor a sufficient condition for uniqueness when the constant correction term is used as a control vector. This means that in some cases the set of observations may contain enough information to specify uniquely the initial state but not the correction term, and vice versa. We showed that if both the initial state and the correction term are used as control vectors, a necessary but not sufficient condition for uniqueness is that conditions for a unique solution using each of the control vectors individually hold. In the time invariant case, we showed that a necessary and sufficient condition for uniqueness using both control vectors is that a full set of observations is available. In each case, adding a background estimate of the control vector to the cost function guarantees uniqueness. These results could be applied more widely in control theory in cases where we wish to determine a constant input from the outputs.

In Chapter 6, we addressed the question of how to allow for a more general form of model error in 4D data assimilation, and in 4D variational assimilation in particular. We considered a general, stochastic representation of model error consisting of serially correlated and serially uncorrelated components. The different representations of model error that have been suggested for use in data assimilation can be expressed using this general form. We considered the technique of state augmentation for estimating the serially correlated component of model error along with the model state in the context of data assimilation, and formulated a general least squares problem for data assimilation allowing for serially correlated model error. This formalism allows us to interpret the correction term technique in a stochastic sense as a method for estimating a constant model bias.

We suggested a "generalized correction term technique" in which the serially uncorrelated part of the model error is neglected, and the augmented initial state is used as an augmented control vector. The generalized correction term technique can therefore allow for various different forms of serially correlated model error. In particular, it can allow for model error which evolves as the model state does. The

theory we present also allows for the dimension m of the correction term to be less than the dimension n of the model state, which could reduce the expense of the assimilation if the effects of model error are known to be localized to a certain area.

As well as considering theoretical aspects of accounting for model error in variational assimilation, we carried out experiments using the correction term technique and generalized correction term technique with simple models exhibiting different types of model error.

In Chapter 5 we compared the initial state, the constant correction term and both together as control vectors in a heat equation model in which model error was due to the omission of a constant source term. Using the correction term as a control vector compensates very well for this model error, and using the correction term in an ensuing forecast gives very good results. We also noted that using the initial state as a control vector partially compensates for the effects of model error, and that using the correction term as the control vector, it is possible to compensate to some extent for errors in the initial state. Using both control vectors together is very effective in this example if we have model error and an unknown initial state. However, this requires about four times as many iterations of the descent algorithm as when only one of the control vectors is used.

In these experiments we also investigated the impact of using a background estimate of zero for the correction term in the cost function, with different values of the weighting q. We found that if the correction term is expected to correct for constant model error, best results are obtained with a small value of q. However, if the correction term is being used to compensate for errors resulting from a wrong initial state, it is important to use a large value of q. We also tried using a correction term with dimension less than that of the model state. Concentrating the correction term around the source point produced good results in far fewer iterations than before.

In Chapter 6, the simple model we used was the linear advection equation with the upwind scheme discretisation. The model error in this example is due to severe dissipation. Using the constant correction term as a control vector has no impact on the model error, which travels across the domain. We tried instead using the

generalized correction term technique allowing the correction term to evolve as the model state does. This successfully reduces the effects of model error during the assimilation interval and also in a subsequent forecast. As found in the experiments of Chapter 5, using the initial state can also compensate to some extent for the effects of model error during the assimilation interval, and the evolving correction term can compensate to some extent for the effects of errors in the initial state during the assimilation interval. Also as in Chapter 5, using both control vectors together compensated successfully for the effects of model error and errors in the initial state. Again, however, many more iterations were required in this case.

Using an evolving correction term is more expensive than using a constant correction term, since an extra set of state and adjoint equations must be integrated. However, these simple experiments demonstrate that using the evolving correction term as a control vector could compensate for the effects of model error which are likely to evolve with the model solution.

In Chapter 7 we carried out similar experiments for a 1D nonlinear shallow water model, using the initial state, a constant correction term and both together as control vectors. We compared the performance of the different control vectors in compensating for errors in the initial state, and for two types of model error: omission of topography and omission of rotation.

As in the earlier experiments, we found that using the initial state as the control vector can compensate to some extent for the effects of model error and can produce a solution closer to the true solution in the middle and at the end of the assimilation interval than if no assimilation were carried out. It is interesting that this is so even though we do not explicitly allow for model error. In each case, however, using a constant correction term as the control vector better compensates for the effects of model error over the assimilation interval. The model error in each case depends on the model state, and motion propagates half way across the model domain in each direction during the assimilation interval. Hence, we concluded that in these experiments the constant correction term is able to compensate for model error depending on the model state on a significant timescale. Using the correction term as a control vector it is also possible to compensate to some extent for the effects of

errors in the initial state, as we noted from the earlier experiments.

We also checked that the correction term recovered in the assimilation was not unreasonably large, and concluded that it did seem to represent an average of the model error source over the assimilation interval. This is important to check, since when using the correction term technique we need to ensure that we are modifying the model equations in a way that makes sense [87].

Using both control vectors together is successful in reducing the large errors caused by both a wrong initial state and model error. In each case the reduction of these errors is greater than if either control vector is used alone, but at the cost that many more iterations of the descent algorithm are required.

The impact of using different values of q, or of how strongly the correction term is constrained to be small in the cost function, is as found in the experiments of Chapter 5. When fewer observations are available, however, using a value of q large enough for smooth solutions produced disappointingly inaccurate results.

Finally, we checked whether the improvement in the solution at the end of the assimilation interval would result in an improvement to a forecast started at this time. We found that it is important not to suddenly cut out the correction term at the beginning of the forecast, and that including the correction term all through a long forecast might have a detrimental impact. Gradually phasing out the correction term during the forecast interval would probably give better results, but further work is needed to check this. In these experiments, however, we found that using the correction term as the control vector to compensate for model error in the assimilation interval, it is possible to obtain a better forecast than is obtained using the initial state as the control vector.

Suggestions for further work

One of the immediate questions arising from this work is that of how to reduce the large number of iterations needed in the minimization procedure using more than one control vector. This could probably be achieved by suitable preconditioning, for example as attempted by D. Zupanski [92], who used serially uncorrelated components of model error as control vectors. Our pleasing results using more than one

control vector indicate that this problem of reducing the number of iterations to an acceptable level is well worth addressing. A second area of our work requiring more attention is that of obtaining smoother solutions when fewer observations are used. This problem is generally tackled by using a background estimate of the control vectors, or extra conditions on smoothness in the cost function.

It would be interesting to carry out more experiments on the shallow water model using other control vectors. An evolving correction term evolving with a simplified, linearized version of the model, perhaps at lower resolution, could be used, perhaps in addition to a constant correction term. Also, it would be worth trying the piecewise constant correction term again.

On a theoretical level, the results given in Chapter 5 on uniqueness and observability could be generalized to allow for an evolving rather than a constant correction term. A major theoretical problem is that of how to specify the model error covariance matrix Q. This is a difficult problem, and affects any attempt to use a stochastic representation for model error in data assimilation [25].

Our work has concentrated on theoretical aspects of the correction term technique and generalizations of it, with tests on simple models. A natural extension to this work would be examine to what extent the same conclusions hold in the context of models which are used operationally. Apart from operational applications, however, the techniques explored here have relevance to other applications to data assimilation, such as estimation of model bias, and in model development. Recently, 4D variational assimilation has been applied to atmospheric chemistry [27]. In this case there may be a very plentiful set of data, but knowledge about the chemical processes constituting the model is incomplete. Here, the generalized correction term technique could be used to indicate where the model is prone to error, and so to use the observational data to infer further information about the model processes.

Bibliography

[1] L. Amodei. Solution approchée pour un problème d'assimilation de données météorologiques avec prise en compte de l'erreur de modèle. *C. R. Acad. Sci. Paris, t.321, série IIa*, pages 1087–1094, 1995.

[2] S. Barnett. *Introduction to Mathematical Control Theory.* OUP, 1975.

[3] G.P. Beaumont. *Probability and Random Variables.* Ellis Harwood.

[4] A.F. Bennett. *Inverse Methods in Physical Oceanography.* CUP, 1992.

[5] A.F. Bennett, B.S. Chua, and L.M. Leslie. Generalized inversion of a global numerical weather prediction model. 1995. Submitted to Meteorology and Atmospheric Physics.

[6] A.F. Bennett, L.M. Leslie, C.R. Hagelberg, and P.E. Powers. Tropical cyclone prediction using a barotropic model initialized by a generalized inverse method. *Monthly Weather Review*, 121:1714–1729, Jun 1993.

[7] A.F. Bennett and P. C. McIntosh. Open ocean modeling as an inverse problem: Tidal theory. *Jounal of Physical Oceanography*, 12:1004–1018, Oct 1982.

[8] A.F. Bennett and R.N. Miller. Weighting initial conditions in variational assimilation schemes. *Monthly Weather Review*, 119:1099–1102, 1991.

[9] P. Bergthorson and B.R. Doos. Numerical weather map analysis. *Tellus*, 7:329–340, 1955.

[10] D.P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods.* Academic Press, 1982.

[11] M. Berz, C. Bischof, G.F. Corliss, and A. Greiwank. *Computational Differentiation: Techniques, Applications and Tools*. SIAM, 1996.

[12] G.J. Boer. A spectral analysis of predictability and error in an operational forecast system. *Monthly Weather Review*, 112:1183–1197, Jun 1984.

[13] F. Bouttier. A dynamical estimation of forecast error covariances in an assimilation system. *Monthly Weather Review*, 122:2376–2390, Oct 1994.

[14] D.E. Catlin. *Estimation, Control, and the Discrete Kalman Filter*. Springer-Verlag, 1989.

[15] W.C. Chao and L-P. Chang. Development of a four-dimensional variational analysis system using the adjoint method at GLA. Part 1: Dynamics. *Monthly Weather Review*, 120:1661–1673, Aug 1992.

[16] S.E. Cohn. An introduction to estimation theory. 1995. Submitted to Journal of the Meteorological Society of Japan.

[17] S.E. Cohn and D. P. Dee. Observability of discretized partial differential equations. *SIAM J. Numer. Anal.*, 25:586–617, Jun 1988.

[18] P. Courtier and O. Talagrand. Variational assimilation of meteorological observations with the adjoint vorticity equation. II. Numerical results. *Quart. J. R. Met. Soc.*, 113:1329–1347, 1987.

[19] P. Courtier and O. Talagrand. Variational assimilation of meteorological observations with the direct and adjoint shallow–water equations. *Tellus*, 42A:531–549, 1990.

[20] P. Courtier, J-N. Thépaut, and A. Hollingsworth. A strategy for operational implementation of 4D-VAR, using an incremental approach. *Q. J. R. Meteorol. Soc.*, 120:1367–1387, 1994.

[21] G.P. Cressman. An operational objective analysis system. *Monthly Weather Review*, pages 367–374, Oct 1959.

[22] A. Dalcher and E. Kalnay. Error growth and predictability in operational ECMWF forecasts. *Tellus*, 39A:474–491, 1987.

[23] R. Daley. The effect of serially correlated observation and model error on atmospheric data assimilation. *Monthly Weather Review*, 120:164–177, Jan 1992.

[24] R.A. Daley. *Atmospheric Data Analysis*. CUP, 1991.

[25] D.P. Dee. On-line estimation of error covariance parameters for atmospheric data assimilation. *Monthly Weather Review*, 123:1128–1145, Nov 1995.

[26] J.C. Derber. A variational continuous assimilation technique. *Monthly Weather Review*, 117:2437–2446, Nov 1989.

[27] M. Fisher and D. J. Lary. Lagrangian four–dimensional variational data assimilation of chemical species. *Q. J. R. Meteorol. Soc.*, 121:1681–1704, 1995.

[28] L.R. Fletcher, J. Kautsky, and N.K. Nichols. Eigenstructure assignment in descriptor systems. *IEEE Trans. Auto. Cnt.*, AC–31, 1986.

[29] R. Fletcher. *Constrained Optimization*. Wiley, 1981.

[30] L. S. Gandin. Objective analysis of meteorological fields. *Gidrometeorol. Izd. Leningrad (in Russian), (English translation by Israel Program for Scientific Translations, Jerusalem, 1965)*, 1963.

[31] A. Gelb. *Applied Optimal Estimation*. MIT Press, Cambridge, Massachusetts, 1974.

[32] M. Ghil and P. Malanotte-Rissoli. Data assimilation in meteorology and oceanography. *Adv. Geophys*, 33:141–266, 1991.

[33] J. C. Gilbert and C. Lemaréchal. Some numerical experiments with variable–storage quasi–Newton algorithms. *Mathematical Programming*, 45:407–435, 1989.

[34] J.C. Gilbert and C. Lemaréchal. The modules M1QN3 and N1QN3. Program documentation, INRIA, 1993.

[35] B. Gilchrist and G.P. Cressman. An experiment in objective analysis. *Tellus*, 6(4):309–318, 1954.

[36] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization.* Academic Press, 1981.

[37] A. Greiwank and Corliss G.F. *Automatic Differentiation of Algorithms.* SIAM, 1991.

[38] A.K. Griffith. Investigation of a simple scheme for data assimilation in a storm surge model. Internal Document 34, Proudman Oceanographic Laboratory, 1992.

[39] A.K. Griffith and N.K. Nichols. Data assimilation using observers. Numerical Analysis Report 11/94, The University of Reading, 1994.

[40] A.K. Griffith and N.K. Nichols. Data assimilation using optimal control theory. Numerical Analysis Report 10/94, The University of Reading, 1994.

[41] A.K. Griffith and N.K. Nichols. Accounting for model error in data assimilation using adjoint models. In *Computational Differentiation: Techniques, Applications and Tools*, pages 195–204. SIAM, 1996. Proceedings of the Second International SIAM Workshop on Conputational Differentiation, Sante Fe, New Mexico, February 1996.

[42] J. Guo and A. da Silva. Computational aspects of Goddard's physical-space statistical analysis system (PSAS). In *Second UNAM-CRAY Supercomputing Conference on Numerical Simulations in the Environmental and Earth Sciences*, Mexico City, Mexico, 1995.

[43] L. Hasdorff. *Gradient Optimization and Nonlinear Control.* Wiley, 1976.

[44] A.H. Jazwinski. *Stochastic processes and filtering theory.* Academic Press, 1970.

[45] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME series D*, 82:35–44, 1960.

[46] R. E. Kalman and R.S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME series D*, 83:35–44, 1961.

[47] J. Kautsky, N.K. Nichols, and P. van Dooren. Robust pole assignment in linear state feedback. *Int. J. Control*, 41(5):1129–1155, 1993.

[48] J-F. Lacarra and O. Talagrand. Short-range evolution of small perturbations in a barotropic model. *Tellus*, 40A:81–95, 1988.

[49] A. Lawless. A perturbation forecast model and its adjoint. In *American Meteorological Society* 11^{th} *Conference on Numerical Weather Prediction*, pages 128–129, Norfolk, Virginia, 1996.

[50] F-X. Le Dimet and M. Ouberdous. Retrieval of balanced fields: An optimal control method. *Tellus*, 45A:449–461, 1993.

[51] F-X. Le Dimet and O. Talagrand. Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, 38A:97–110, 1986.

[52] J.M. Lewis and J.C.Derber. The use of adjoint equations to solve a variational adjustment problem with advective constraints. *Tellus*, 37A:309–322, 1985.

[53] J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer–Verlag, 1971. (English translation).

[54] A.C. Lorenc. A global three–dimensional multi–variate statistical interpolation scheme. *Monthly Weather Review*, 109:701–721, 1981.

[55] A.C. Lorenc. Analysis methods for numerical weather prediction. *Q.J.R. Meteorol. Soc.*, 112:1177–1194, 1986.

[56] A.C. Lorenc. Optimal nonlinear objective analysis. *Q.J.R. Meteorol. Soc.*, 114:205–240, 1988.

[57] A.C. Lorenc. Iterative analysis using covariance functions and filters. *Q.J.R. Meteorol. Soc.*, 118:569–591, 1992.

[58] A.C. Lorenc and O. Hammon. Objective quality control of observations using Bayesian methods. Theory, and a practical implementation. *Q.J.R. Meteorol. Soc.*, 114:515–543, 1988.

[59] B. Machenhauer. On the dynamics of gravity oscillations in a shallow water model, with applications to normal mode initialization. *Beitraege zur Physik der Atmosphaere*, 50:253–271, 1977.

[60] R Ménard. *Kalman Filtering of Burgers' equation and its application to atmospheric data assimilation.* PhD thesis, McGill University, 1994.

[61] R. Ménard and R. Daley. The application of Kalman smoother theory to the estimation of 4DVAR error statistics. *Tellus*, 48A:221–237, 1996.

[62] H. Michalska. An observer for nonlinear descriptor systems of index one. In *Proceedings of the International Symposium on Implicit and Nonlinear Systems*, pages 372–379, 1992.

[63] R.N. Miller, M. Ghil, and F. Gauthiez. Advanced data assimilation in strongly nonlinear dynamical systems. *Journal of the Atmospheric Sciences*, 51(8):1037–1056, 1994.

[64] P.J. Moylan. Stable inversion of linear systems. *IEEE Trans. on Autom. Control*, 22:74–78, 1977.

[65] J. Nocedal. Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, 35(151):773–782, 1980.

[66] J. O'Reilly. *Observers for Linear Systems.* Academic Press, 1983.

[67] S. Oren and E. Spedicato. Optimal conditioning of self-scaling variable metric algorithms. *Mathematical programming*, 10:70–90, 1976.

[68] H.A. Panofski. Objective weather-map analysis. *Journal of Meteorology*, 6:386–392, 1949.

[69] C.A. Parrett and M.J.P. Cullen. Simulation of hydraulic jumps in the presence of rotation and mountains. *Quart. J. R. Met. Soc.*, 110:147–165, 1984.

[70] R.V. Patel. Minimal-order inverses for linear systems with zero and arbitrary initial states. *Int. J. Control*, 30(2):245–258, 1979.

[71] R.V. Patel. Construction of stable inverses for linear systems. *Int. J. Systems Sci.*, 13(5):499–515, 1982.

[72] C. Pires and O. Talagrand. On extending the limits of variational assimilation in nonlinear chaotic systems. *Tellus*, 48A:96–121, 1996.

[73] F. Rabier, P. Courtier, J. Pailleux, O. Talagrand, and D. Vasiljevic. A comparison between four-dimensional variational assimilation and simplified sequential assimilation relying on three-dimensional variational analysis. *Q. J. R. Meteorol. Soc.*, 119:845–880, 1993.

[74] J. Rinne and H. Jarvinen. Estimation of the Cressman term for a barotropic model through optimization with use of the adjoint method. *Monthly Weather Review*, 121:825–833, 1993.

[75] Y. Sasaki. An objective analysis based on the variational method. *Journal of the Meteorological Society Japan*, 36(3):77–88, 1958.

[76] Y. Sasaki. Numerical variational analysis formulated under the constraints as determined by longwave equations and a low-pass filter. *Monthly Weather Review*, 98:884–898, 1970.

[77] Y. Sasaki. Numerical variational analysis with weak constraint and application to surface analysis of severe storm gust. *Monthly Weather Review*, 98:899–910, 1970.

[78] Y. Sasaki. Some basic formalisms in basic variational analysis. *Monthly Weather Review*, 98:875–883, 1970.

[79] S.M. Stringer. *The Use of Robust Observers in the Simulation of Gas Supply Networks*. PhD thesis, The University of Reading, Department of Mathematics, 1993.

[80] O. Talagrand and P. Courtier. Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. *Quart. J. R. Met. Soc.*, 113:1311–1328, 1987.

[81] A. Tarantola. *Inverse problem theory. Methods for data fitting and model parameter estimation*. Elsevier, 1987.

[82] W.C. Thacker. Relationships between statistical and deterministic methods of data assimilation. In *Variational Methods in the Geosciences*. Elsevier, 1986.

[83] W.C. Thacker and R.B. Long. Fitting dynamics to data. *Journal of Geophysical Research*, 93(c2):1227–1240, 1988.

[84] R. Todling and S. E. Cohn. Suboptimal schemes for atmospheric data assimilation based on the Kalman filter. *Monthly Weather Review*, 122:2530–2557, 1994.

[85] J.J. Tribbia and D.P. Baumhefner. The reliability of improvements in deterministic short–range forecasts in the presence of initial state and modeling deficiencies. *Monthly Weather Review*, 108:2276–2288, 1988.

[86] L. Weiss. Controllability, realization and stability of discrete-time systems. *SIAM J. Control*, 10(2):230–251, 1972.

[87] W. Wergen. The effect of model errors in variational assimilation. *Tellus*, 44A:297–313, 1992.

[88] J.H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, 1965.

[89] M.A. Wolfe. *Numerical Methods for Unconstrained Optimization, an Introduction*. Van Nostrand Reinhold Company, 1978.

[90] W. Wonham. On pole assignment in multi–input, controllable linear systems. *IEEE Trans Automatic Control*, AC(12):660–665, 1967.

[91] X. Zou, I.M. Navon, and F-X. Le Dimet. Incomplete observations and control of gravity waves in variational assimilation. *Tellus*, 44A:273–296, 1992.

[92] D. Zupanski. A general weak constraint applicable to operational 4DVAR data assimilation systems. 1996. Submitted to Monthly Weather Review.

[93] D. Zupanski and F. Mesinger. Four–dimensional variational data assimilation of precipitaion data. *Monthly Weather Review*, 123:1112–1127, 1995.

[94] M. Zupanski. Regional four–dimensional variational data assimilation in a quasi–operational forecasting environment. *Monthly Weather Review*, 121:2396–2408, 1993.