

THE UNIVERSITY OF READING
DEPARTMENT OF MATHEMATICS

**Joint state and parameter estimation
using data assimilation
with application to
morphodynamic modelling**

Polly J. Smith

Thesis submitted for the degree of
Doctor of Philosophy
June 2010

Abstract

Data assimilation is often used to provide initial conditions for state estimation; combining model predictions with observational data to produce an updated model state that most accurately characterises the true system state whilst keeping the model parameters fixed. This updated model state is then used to initiate the next model forecast. However, even with perfect initial data, inaccurate model parameters will lead to the growth of prediction errors. A fundamental question in model development is how to estimate parameters a priori.

In this thesis we investigate the application of data assimilation to model parameter estimation. By employing the technique of state augmentation we develop a novel framework for estimating uncertain model parameters concurrently with the model state.

A key difficulty in data assimilation is specification of the background error covariances. For combined state-parameter estimation, it is important that the cross covariances between the parameters and the state are given a good a priori specification. We find that in order to yield reliable estimates of the true parameters, a flow dependent representation of the state-parameter cross covariances is required. By combining ideas from 3D-Var and the extended Kalman filter we develop a new hybrid assimilation algorithm that captures the flow dependent nature of the state-parameter cross covariances without the complexity and computational expense of explicitly propagating the full system covariance matrix.

We demonstrate the applicability of the method to a range of simple models. Ideas are developed in the context of morphodynamic modelling but the scheme is also shown to be effective in both linear and nonlinear general dynamical system models, recovering the true model parameter values to a good level of accuracy.

Declaration

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Polly Smith

Acknowledgements

Firstly, I'd like to thank my supervisors Dr Sarah Dance and Prof. Nancy Nichols for their guidance, support and endless enthusiasm over the past three and a half years; they have given me opportunities I could never have imagined. I would also like to thank Prof. Mike Baines for his assistance in the early stages of the project and for helping to persuade me that quitting the world of government statistics and returning to academic life was a good idea in the first place. I am also grateful to everyone involved in the FREE Changing Coastlines project for useful discussions and feedback and to HR Wallingford for their time and interest.

An huge thank you to my family for their encouragement and support and for taking care of me during the various foot related sagas! Thank you to all my friends for keeping me sane. A special mention to the girls, for always being there and for offering distraction during times of meltdown.

Finally, I acknowledge the support of the National Environmental Research Council (NERC) and the UK Environment Agency.

Contents

1	Introduction	1
1.1	Main aims	3
1.2	Summary of new results	3
1.3	Outline of thesis	4
2	Data assimilation	6
2.1	Nonlinear model system equations	6
2.2	Linear model system equations	8
2.3	Data assimilation methods	8
2.3.1	3D-Var and Optimal Interpolation	9
2.3.2	The Kalman filter	12
2.3.3	The Extended Kalman Filter	14
2.3.4	Practical implementation	15
2.4	Summary	16
3	Parameter estimation	17
3.1	Ill-posedness, identifiability and uniqueness	18
3.2	Model calibration	19
3.2.1	Model calibration in coastal modelling	20
3.2.2	Discussion	22
3.3	Dual state-parameter estimation	23
3.4	State augmentation	24
3.4.1	Applications of state augmentation	25
3.4.2	Mathematical formulation of the augmented system	26
3.5	Summary	30

4	Morphodynamic modelling and data assimilation	31
4.1	Morphodynamic modelling	31
4.2	Modelling the sediment transport flux	32
4.3	Observation sources	35
4.4	Morphodynamics and data assimilation	36
4.5	Summary	37
5	System development	38
5.1	Linear advection	39
5.1.1	Discretisation	40
5.1.2	Parameter model	42
5.2	The minimisation algorithm	43
5.2.1	Verification	44
5.3	Error covariances	44
5.3.1	Observation error covariance	45
5.3.2	State background error covariance	45
5.3.3	Experiments	46
5.3.4	Results	48
5.4	Joint state-parameter estimation	52
5.4.1	Parameter error covariance	53
5.4.2	Modelling the state-parameter cross covariances	53
5.4.3	State background errors	54
5.4.4	State-parameter cross covariance	55
5.5	Assimilation experiments	56
5.5.1	True model state	57
5.5.2	Observations	57
5.5.3	Background error covariance	58
5.5.4	Results	58
5.6	An alternative approach	60
5.6.1	Results	62
5.7	Summary	65
6	A hybrid approach	66
6.1	Formulation	67

6.2	Practical implementation	70
6.3	Discussion	71
6.4	Hybrid sequential data assimilation	72
6.5	Summary	73
7	Hybrid assimilation experiments	74
7.1	Linear advection	74
7.1.1	State-parameter cross covariance	76
7.1.2	Assimilation experiments	77
7.1.3	Results	79
7.2	Nonlinear oscillator	88
7.2.1	State-parameter cross covariance	89
7.2.2	Assimilation experiments	90
7.2.3	Results	91
7.3	The Lorenz 63 equations	107
7.3.1	State-parameter cross covariance	109
7.3.2	Assimilation experiments	110
7.3.3	Results	110
7.4	Summary	123
8	1D sediment transport model	125
8.1	The model	125
8.2	Discretisation of the state evolution model	127
8.3	The parameter model	129
8.4	State-parameter cross covariances and parameter covariances	130
8.5	Assimilation experiments	133
8.6	Results	135
8.6.1	Perfect observations	135
8.6.2	Discussion	136
8.6.3	Noisy observations	142
8.7	Summary	144
9	Conclusions and further work	149
9.1	Conclusions	149
9.2	Further work	154

A	Tangent Linear Model (TLM)	157
A.1	Definition	157
A.2	Tangent Linear of the augmented system model	157
B	Block Matrix Inversion	159
C	The Upwind scheme for the linear advection equation	160
C.1	Model validation	160
D	Jacobian matrix calculations	162
D.1	Nonlinear oscillating system	162
D.2	Lorenz equations	163
E	SLCN scheme validation	166
E.1	The semi-Lagrangian scheme	166
E.2	The Crank-Nicolson scheme	167
E.3	Combined semi-Lagrangian Crank-Nicolson scheme	168

List of Figures

5.1	Solutions to the linear advection equation for Gaussian initial data. . . .	40
5.2	Comparison of analytic and numerical solutions to the linear advection equation using the semi-Lagrangian method.	42
5.3	The quasi-Newton method applied to Rosenbrock's function.	44
5.4	Comparison of diagonal, tri-diagonal, Gaussian and Markov state background error covariance matrices.	49
5.5	Effect of varying the correlation length scale in the Gaussian matrix. . .	51
5.6	Effect of varying the correlation length scale in the Markov matrix. . . .	52
5.7	Linear advection model solution with unknown initial bathymetry and unknown advection velocity.	55
5.8	A Gaussian function and its derivative.	58
5.9	Linear advection model: assimilation with static \mathbf{B}_{zp} , state analysis for initial parameter estimate $\tilde{c} = 0.004$ m/s.	59
5.10	Linear advection model: assimilation with static \mathbf{B}_{zp} , updating of parameter c	59
5.11	Linear advection model: assimilation with static \mathbf{B}_{zp} , state analysis for initial parameter estimate $\tilde{c} = 0.02$ ms ⁻¹	60
5.12	Linear advection model: assimilation with time varying \mathbf{B}_{zp} , updating of parameter c	63
5.13	Linear advection model: assimilation with time varying \mathbf{B}_{zp} , state analysis for initial parameter estimate $\tilde{c} = 0.004$ ms ⁻¹	63
5.14	Linear advection model: assimilation with time varying \mathbf{B}_{zp} , updating of parameter c	64
5.15	Linear advection model: assimilation with time varying \mathbf{B}_{zp} , updating of parameter c	64

5.16	Linear advection model: assimilation with time varying \mathbf{B}_{zp} , state analysis for initial parameter estimate $\tilde{c} = 0.02 \text{ ms}^{-1}$	65
7.1	Linear advection model: state analysis with perfect observations assimilated every $25\Delta x, 10\Delta t$	81
7.2	Linear advection model: varying the spatial frequency of observations, parameter updates for initial estimate $c = 0.87116$	82
7.3	Linear advection model: varying the temporal frequency of observations, parameter updates for initial estimate $c = 0.87116$	82
7.4	Linear advection model: state analysis with perfect observations assimilated every $50\Delta x, 10\Delta t$	83
7.5	Linear advection model: state analysis with perfect observations assimilated every $50\Delta t, 10\Delta x$	84
7.6	Linear advection model: noisy observations, parameter updates for initial estimate $c = 0.87116$	85
7.7	Linear advection model: state analysis with noisy observations assimilated every $10\Delta x, 10\Delta t, \sigma_o^2 = 0.01$	86
7.8	Linear advection model: state analysis with noisy observations assimilated every $10\Delta x, 10\Delta t, \sigma_o^2 = 0.1$	87
7.9	Nonlinear oscillator: computed numerical solution for x and y	91
7.10	Nonlinear oscillator: varying the temporal frequency of observations, parameter updates example (i).	93
7.11	Nonlinear oscillator: varying the temporal frequency of observations, parameter updates example (ii).	94
7.12	Nonlinear oscillator experiments: analysis for x and y , example (i) with perfect observations assimilated every $25\Delta t$	95
7.13	Nonlinear oscillator experiments: analysis for x and y , example (i) with perfect observations assimilated every $50\Delta t$	96
7.14	Nonlinear oscillator: analysis for x and y , example (i) with perfect observations assimilated every $50\Delta t$, inflated σ_b^2	97
7.15	Nonlinear oscillator: noisy observations, parameter updates for example (i), $\sigma_o^2 = 0.01$	98
7.16	Nonlinear oscillator: noisy observations, averaged parameter updates for example (i), $\sigma_o^2 = 0.01$	99

7.17	Nonlinear oscillator: analysis for x and y , example (i) with noisy observations assimilated every $50\Delta t$, $\sigma_o^2 = 0.01$	100
7.18	Nonlinear oscillator: noisy observations, parameter updates for example (i), $\sigma_o^2 = 0.1$	101
7.19	Nonlinear oscillator: analysis for x and y , example (i) with noisy observations assimilated every $10\Delta t$, $\sigma_o^2 = 0.1$	102
7.20	Nonlinear oscillator: analysis for x and y , example (i) with noisy observations assimilated every $25\Delta t$, $\sigma_o^2 = 0.1$	103
7.21	Nonlinear oscillator - overweighting the observations: analysis for x and y , example (i) with noisy observations assimilated every $25\Delta t$	104
7.22	Nonlinear oscillator - overweighting the observations: parameter updates for example (i).	105
7.23	Lorenz equations: computed numerical solution for x and z	111
7.24	Lorenz equations: varying the temporal frequency of observations, parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$	112
7.25	Lorenz equations: analysis for x and z with perfect observations assimilated every $30\Delta t$	113
7.26	Lorenz equations: analysis for x and z with perfect observations assimilated every $40\Delta t$	114
7.27	Lorenz equations: noisy observations, parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$, $\sigma_o^2 = 0.1$	115
7.28	Lorenz equations: noisy observations, parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$, $\sigma_o^2 = 0.25$	116
7.29	Lorenz equations: noisy observations, averaged parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$, $\sigma_o^2 = 0.1$	117
7.30	Lorenz equations: noisy observations, averaged parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$, $\sigma_o^2 = 0.25$	118
7.31	Lorenz equations: analysis for x and z with noisy observations assimilated every $30\Delta t$, $\sigma_o^2 = 0.1$	119
7.32	Lorenz equations: analysis for x and z with noisy observations assimilated every $30\Delta t$, $\sigma_o^2 = 0.25$	120
7.33	Lorenz equations - time averaged parameter estimates: analysis for x and z with noisy observations assimilated every $30\Delta t$, $\sigma_o^2 = 0.1$	121

8.1	Solutions to the quasi-linear advection equation for Gaussian initial data.	127
8.2	Approximating the matrix \mathbf{N}_k .	131
8.3	1D sediment transport model: varying the temporal frequency of observations, parameter updates example (i).	137
8.4	1D sediment transport model: varying the temporal frequency of observations, parameter updates example (ii).	138
8.5	1D sediment transport model: varying the spatial frequency of observations, parameter updates example (i).	139
8.6	1D sediment transport model: state analysis, example (i) with perfect observations assimilated every $25\Delta x$, $24\Delta t$.	140
8.7	1D sediment transport model: state analysis, example (i) with perfect observations assimilated every $100\Delta x$, $4\Delta t$.	141
8.8	1D sediment transport model: incorrect parameter cross covariance, parameter updates example (i).	143
8.9	1D sediment transport model: noisy observations, parameter updates example (i).	145
8.10	1D sediment transport model: noisy observations, averaged parameter updates example (i).	146
8.11	1D sediment transport model: state analysis, example (ii) with noisy observations assimilated every $25\Delta x$, $4\Delta t$, $\sigma_o^2 = 0.1$.	147
C.1	Comparison of analytic and numerical solutions to the linear advection equation using the upwind method.	161
E.1	Comparison of analytic and numerical solutions to the inviscid Burgers' equation using the semi-Lagrangian method.	167
E.2	Comparison of analytic and numerical solutions to the heat equation using the Crank-Nicolson scheme.	168
E.3	Comparison of numerical and analytical solutions to the viscous Burgers' equation using the SLCN scheme.	169

Acronyms

3D-Var	Three dimensional variational assimilation
4D-Var	Four dimensional variational assimilation
BLUE	Best Linear Unbiased Estimator
CFL	Courant Fredrichs Lewy
EKF	Extended Kalman filter
EnKF	Ensemble Kalman Filter
GPS	Global Positioning System
KF	Kalman filter
GLUE	Generalised Likelihood Uncertainty Estimation
LiDAR	Light Detection and Ranging
MLEF	Maximum likelihood ensemble filter
NWP	Numerical Weather Prediction
OI	Optimal interpolation
OSSE	Observing system simulation experiment
SAR	Synthetic aperture radar
SLCN	Semi-Lagrangian Crank-Nicolson
SODA	Simultaneous optimisation and data assimilation

Data assimilation notation

\mathbf{z}	state vector
\mathbf{z}^t	‘true’ state vector
\mathbf{z}^b	background/ forecast state vector
\mathbf{z}^a	analysis state vector
\mathbf{p}	parameter vector
\mathbf{y}	vector of observations
\mathbf{h}	observation operator
\mathbf{H}	linearised observation operator
\mathbf{f}	nonlinear forward model
\mathbf{M}	linear forward model
\mathbf{M}^T	adjoint model
\mathbf{K}	gain matrix
$\varepsilon_{\mathbf{z}}^b$	state background error
δ	observation error
\mathbf{B}	background error covariance matrix
\mathbf{R}	observation error covariance matrix
σ_b^2	state background error variance
σ_o^2	observation error variance
$\rho = \{\rho_{ij}\}$	correlation matrix
L	background error correlation length scale
J	cost function
m	dimension of the state vector
q	dimension of the parameter vector
r	dimension of the observation vector
\mathbf{P}^f	Kalman filter forecast error covariance matrix
\mathbf{P}^a	Kalman filter analysis error covariance matrix

Augmented system notation

\mathbf{w}	augmented state vector
$\tilde{\mathbf{f}}$	augmented nonlinear forward model
\mathbf{F}	linearised augmented forward model
$\tilde{\mathbf{h}}$	augmented observation operator
$\tilde{\mathbf{H}}$	linearised augmented observation operator
\mathbf{B}_{zz}	state background error covariance matrix
\mathbf{B}_{zp}	state-parameter background error cross covariance matrix
\mathbf{B}_{pp}	parameter background error covariance matrix
$\varepsilon_{\mathbf{p}}^b$	parameter background error vector
$\varepsilon_{\mathbf{w}}^b$	augmented state background error vector
$\mathbf{K}_{\mathbf{z}}$	state gain matrix
$\mathbf{K}_{\mathbf{p}}$	parameter gain matrix
\mathbf{N}	Jacobian of the state forecast model with respect to the parameter vector.

General model notation

Δx	model grid spacing
Δt	model time step
k	discrete time index
j	discrete space index

Linear advection model notation

$z(x, t)$	bed height
c	advection velocity
\tilde{c}	estimated advection velocity
$f(x)$	true initial state
$\tilde{f}(x)$	predicted initial state
α, β, γ	scaling parameters for initial data
σ_c^2	parameter c error variance

Semi-Lagrangian discretisation:

α_j particle displacement

x_d departure point at time level t_k

x_j arrival point at time level t_{k+1}

Upwind discretisation:

$\mu \quad \frac{\Delta t}{\Delta x}$

Nonlinear oscillating system notation

\mathbf{x} model state vector

$x(t)$ displacement

$y(t)$ velocity dx/dt

d model parameter (damping)

m model parameter (square of frequency of oscillation)

Lorenz 63 equations notation

\mathbf{x} model state vector

$x(t), y(t), z(t)$ model variables

σ, ρ, β model parameters

1D sediment transport model notation

A	parameter for calculating the sediment transport rate
$c(z, q)$	bed celerity/ advection velocity
ϵ	sediment porosity
F	water flux
h	water height
κ	diffusion coefficient
q	total load sediment transport rate in the x direction
u	depth-averaged current
n	parameter for calculating the sediment transport rate
$z(x, t)$	bathymetry/ bed height
δA	parameter A perturbation
δn	parameter n perturbation
σ_A^2, σ_n^2	parameter A and n error variances
σ_{An}	parameter A and n cross covariance

SLCN discretisation:

\mathcal{L}	discrete Laplacian operator
\mathcal{C}_*	interpolation operator
\mathbf{x}	vector of regular model spatial grid points
\mathbf{x}_d	vector of departure points
$\boldsymbol{\alpha}$	displacement vector
ν	$\frac{\kappa}{\Delta x^2}$

Chapter 1

Introduction

Numerical models are used for forecasting across a wide range of environmental applications. The aim is to have a model that characterises the behaviour of the system of interest as accurately as possible, whilst satisfying known physical properties. Unfortunately, a mathematical model can never completely describe the complex physical processes underlying a real world dynamical system. Advances in knowledge and computing technology mean that environmental forecasting models are becoming increasingly sophisticated, but in practice these models suffer from uncertainty in their initial conditions and parameters. Even with perfect initial data, inaccurate representation of model parameters will lead to the growth of model error and therefore affect the ability of a model to accurately predict the true system state.

Parameters are intrinsic to environmental modelling. Parameterisations are typically used in applications where the underlying physics of a process are not fully known or understood, or to model subgrid scale effects that cannot be captured within a particular model resolution. Coastal morphodynamic modelling is one such field. Coastal morphodynamics is the study of sediment transport processes occurring in coastal regions. Morphodynamic change can have a great many consequences, both environmental and economic [68]. Knowledge of coastal morphology is fundamental to the effective management of coastal regions; for example, managing coastal erosion, assessing the potential impact of human use of coastal land, monitoring wildlife habitats and the mitigation of flood hazard. It is a complex and challenging subject but one which is of great practical

importance.

Many of the processes governing sediment transport are not fully understood or measurable. This lack of detail in knowledge means that sediment transport models are often based on a mixture of theory and empirical results. As a consequence of this, models contain parameters that are not directly measurable and which must be ‘tuned’ in order to calibrate the model to a specific field site. Poorly known input parameters are a key source of uncertainty in sediment transport models [94]. A fundamental question in model development is how to estimate these parameters a priori. One option is to use data assimilation.

Data assimilation is a sophisticated mathematical technique for combining observational data with model predictions. Whilst data assimilation has been in use in the context of atmospheric and oceanic prediction for some years, it has rarely been employed for morphodynamic modelling. Data assimilation is most commonly used to produce initial conditions for state estimation; estimating model variables whilst keeping the model parameters fixed. However, by employing the method of state augmentation, it is also possible to use data assimilation to estimate uncertain model parameters concurrently with the model state.

State augmentation is a conceptually simple technique that offers a framework for estimating uncertain model parameters jointly with the model state variables as part of the assimilation process [45]. The parameters are appended to the model state vector, the model prediction equations are combined with evolution equations for the parameters and the chosen assimilation algorithm is simply applied to this new augmented system in the usual way [72]. The same approach has previously been successfully used in the treatment of systematic model error or bias (see, for example, [5], [16], [34], [64]).

In theory, state augmentation can be applied with any of the standard data assimilation methods. For the work in this thesis we use a three dimensional variational (3D-Var) data assimilation scheme (e.g. [13], [58]). Variational methods are commonly used for large scale problems. 3D-Var is an established method that has many advantages over other assimilation techniques, such as ease of implementation, computational efficiency and robustness.

As we show here, a key difficulty in the construction of a data assimilation algorithm is specification of the statistics of the background errors. These statistics, in the form of error covariances, play an important role in the filtering and spreading of observational data. For parameter estimation, it is the joint state-parameter cross-covariances that transfer information from the observations to the parameter estimates and therefore play a crucial role in the parameter updating. A good a priori specification of these covariances is vital for accurate parameter updating.

The main aims and principal results of this thesis are summarised in the next two sections. This is followed by an overview of each chapter in section 1.3. Much of the material in this thesis has previously been presented in the departmental reports [88], [89], [91], [92] and also published in the journal paper [90].

1.1 Main aims

The objectives of this thesis are:

- To consider the problem of model parameter estimation in environmental modelling with specific reference to coastal engineering applications.
- To evaluate the potential of state augmentation as an approach to concurrent model state-parameter estimation using sequential 3D-Var data assimilation.
- To examine some of the issues associated with the development of an augmented data assimilation system. In particular, the role of the background error covariances in the assimilation and how these should be defined for joint estimation.
- To present a new method for treatment of the state-parameter cross covariances and to demonstrate the utility of this new algorithm using a range of simple dynamical system models.

1.2 Summary of new results

The main results of this thesis are:

1. By employing the technique of state augmentation we have developed a novel hybrid method for concurrent state-parameter estimation using sequential data assimilation.
2. The success of the state augmentation approach relies heavily on the relationship between the state and parameters (as described by the state-parameter cross covariances) being well defined. The assumption of stationary background error statistics made by standard 3D-Var algorithms for basic state estimation is insufficient for joint state-parameter estimation. Here, we have shown that a flow dependent description of the state-parameter cross covariances is required. However it is not necessary to evolve the full background error covariance matrix.
3. Our unique hybrid assimilation algorithm captures the flow dependence of the state-parameter errors without the computational expense of explicitly propagating the full system covariance matrix. This is achieved by combining a static representation of the state background error covariances with a time varying approximation of the state-parameter cross covariances.
4. For models with more than one parameter, consideration must also be given to the relationship between individual parameters. Where model parameters exhibit strong interdependence it is important that the elements of the parameter background error covariance matrix reflect the degree and direction of correlation.
5. Our new scheme has proved effective in a range of simple linear and nonlinear dynamical system models. We are able to recover the true parameter values to a good level of accuracy, even when observational data are noisy and this can lead to significantly improved forecasts of the state.

1.3 Outline of thesis

We begin chapter 2 by describing the data assimilation problem for state estimation in a general system model. We then discuss the data assimilation methods relevant to the work in this thesis in more detail.

In chapter 3 we discuss the problem of model parameter estimation and review examples

of some of the different approaches to model calibration in coastal modelling applications. We then introduce the technique of state augmentation and explain how data assimilation can be used to estimate poorly known model parameters jointly with the model state.

We give a general introduction to morphodynamic modelling in chapter 4 and explain some of the difficulties associated with the development of an effective coastal morphodynamic model. We describe some of the various sources of observations available for morphodynamic data assimilation and review examples of previous applications of data assimilation to coastal modelling.

The development of our joint state-parameter estimation system is described in chapter 5. This chapter brings together various elements of the work originally reported in [88], [89] and [90]. We use a simple 1D linear advection model to help demonstrate the theory and highlight some of the issues associated with practical implementation of the 3D-Var and state augmentation techniques. We use a series of simple experiments to illustrate the role of the background error covariances and examine ways of defining the state-parameter cross covariances required for concurrent state-parameter estimation.

We consider the question of how to specify the augmented background error covariance matrix for a general system model in chapter 6. We use the results of chapter 5 to establish a novel hybrid algorithm that is applicable to a range of dynamical system models.

In chapter 7 we assess the potential transferrability of this proposed new approach via a series of identical twin experiments with three simple dynamical system models. Additional results from a similar set of experiments can be found in [92].

In chapter 8 we evaluate our new hybrid algorithm in the context of morphodynamic modelling using the 1D nonlinear sediment transport model introduced in chapter 4 and also discussed in [91]. This model has two uncertain, but highly correlated parameters which causes difficulties relating to identifiability and non-uniqueness of solutions.

Finally, in chapter 9 we summarise the conclusions from this work and discuss possible future developments.

Chapter 2

Data assimilation

In this chapter we lay the foundations for understanding the methods we present in subsequent chapters. Since the data assimilation methods we use are relevant in a wide range of contexts we begin by describing the data assimilation problem for state estimation in a general system model. The specific models that we have been using to develop and test our ideas will be introduced in later chapters. In the next section we introduce the general nonlinear model system equations and explain the terminology and notation that we will use throughout this thesis. Equations for a linear model system are given in section 2.2. Then, in section 2.3, we give a brief overview of the main data assimilation methods before discussing the schemes relevant to this work in more detail. In section 2.3.4 we outline some of the issues associated with practical implementation of these schemes. We limit our discussion to discrete, finite dimensional, systems as this is most useful for application to the numerical models in which we are interested. Our notation is similar to that of [44].

2.1 Nonlinear model system equations

We consider the discrete nonlinear time invariant dynamical system model

$$\mathbf{z}_{k+1} = \mathbf{f}(\mathbf{z}_k, \mathbf{p}) \quad k = 0, 1, \dots \quad (2.1)$$

The column vector $\mathbf{z}_k \in \mathbb{R}^m$ is known as the state vector; it contains the model variables at each grid cell in the model domain and represents the model state at time t_k . The

operator $\mathbf{f} : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is a nonlinear function describing the evolution of the state from time t_k to t_{k+1} and $\mathbf{p} \in \mathbb{R}^q$ is a vector of q (uncertain) model parameters.

We assume that specification of the model state and parameters at time t_k uniquely determines the model state at all future times. We also assume that $\mathbf{f}(\mathbf{z}, \mathbf{p})$ is differentiable with respect to \mathbf{z} and \mathbf{p} for all $\mathbf{z} \in \mathbb{R}^m$ and $\mathbf{p} \in \mathbb{R}^q$. Later in this thesis we consider the case where the model state vector \mathbf{z} is a 1D vector representing bathymetry or bed height and the operator $\mathbf{f}(\mathbf{z}, \mathbf{p})$ represents the equations describing the evolution of the bed-form over time.

We suppose that, at time t_k , we have a set of r_k observations to assimilate and that these are related to the model state by the equations

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{z}_k) + \boldsymbol{\delta}_k, \quad k = 0, 1, \dots \quad (2.2)$$

Here $\mathbf{y}_k \in \mathbb{R}^{r_k}$ is a vector of r_k observations at time t_k . Note that the number of available observations r_k may vary with time. The vector $\boldsymbol{\delta}_k \in \mathbb{R}^{r_k}$ represents the observation errors. These errors are commonly assumed to be unbiased, serially uncorrelated, stochastic variables, with a given probability distribution [54]. The operator $\mathbf{h}_k : \mathbb{R}^m \rightarrow \mathbb{R}^{r_k}$ is a nonlinear observation operator that maps from model to observation space, converting the model variables to model values of the observed variable. If, for example, we have direct measurements of bed height but at points that do not coincide with the model grid, \mathbf{h} is simply an interpolation operator that interpolates the model bathymetry from the model grid to the observation locations. Often, the model variables we wish to analyse cannot be observed directly and instead we have observations of another measurable quantity. In this case \mathbf{h} will also include transformations based on physical relationships that convert the model variables to the observations.

We also suppose that we have a *background* state $\mathbf{z}_k^b \in \mathbb{R}^m$, with error $\boldsymbol{\varepsilon}_{\mathbf{z}_k^b} \in \mathbb{R}^m$, that represents an *a priori* estimate of the true system state \mathbf{z}_k^t at time t_k . This is a best guess estimate of the current system state obtained (for example) from a previous model forecast.

The true system state is denoted by the vector $\mathbf{z}_k^t \in \mathbb{R}^m$. We can write the evolution of the true state in terms of the model (2.1) as

$$\mathbf{z}_{k+1}^t = \mathbf{f}(\mathbf{z}_k^t, \mathbf{p}^t) + \boldsymbol{\varepsilon}_k \quad k = 0, 1, \dots \quad (2.3)$$

where the vector $\boldsymbol{\varepsilon}_k$ represents the model error. For the work in this thesis we set $\boldsymbol{\varepsilon}_k = 0$. This is known as the ‘perfect model’ assumption [75]. In other words, we assume that for any given initial state the model equations (2.1), with true parameters \mathbf{p}^t , give a ‘perfect’ representation of the true dynamical system state on the model grid.

For state estimation, the aim of data assimilation is to combine the measured observations \mathbf{y}_k with the model predictions \mathbf{z}_k^b in order to derive an updated model state that most accurately describes the true system state \mathbf{z}_k^t at time t_k . This optimal estimate is called the *analysis* and is denoted \mathbf{z}_k^a . The model parameters \mathbf{p} are assumed to be known constant values. We describe the data assimilation problem for combined state-parameter estimation later in chapter 3, section 3.4.

2.2 Linear model system equations

We will also consider examples in which the forecast model and/ or observations are linear functions of the model state. In this case the model equations (2.1) and (2.2) can be written as

$$\mathbf{z}_{k+1} = \mathbf{M}_k(\mathbf{p}) \mathbf{z}_k \quad k = 0, 1, \dots \quad (2.4)$$

and

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{z}_k + \boldsymbol{\delta}_k, \quad k = 0, 1, \dots \quad (2.5)$$

where the matrix $\mathbf{M}_k \in \mathbb{R}^{m \times m}$ is a non-singular matrix that depends nonlinearly on the parameters \mathbf{p} , and $\mathbf{H}_k \in \mathbb{R}^{r \times m}$.

2.3 Data assimilation methods

There are many different types of data assimilation algorithm, each varying in formulation, complexity, computational burden, optimality and suitability for practical application. A popular distinction is between three and four dimensional schemes. Three dimensional algorithms consider only the spatial estimation problem, that is, observations are distributed in space but treated at a single point in time. Four dimensional techniques incorporate the time dependency of observations, using the model dynamics

to produce an analysis state that gives the best fit to observations taken over a given time window.

Here we focus our attention on the particular schemes upon which the work presented in this thesis is based. These are: three dimensional variational data assimilation (3D-Var), the Kalman filter (KF) and the extended Kalman filter (EKF). For further details of the various other data assimilation schemes we refer the reader to the literature. A useful overview of some of the most common data assimilation methods used in meteorology and oceanography are given the review articles by Ghil and Malanotte-Rizzoli [27] and Lorenc [57]. More more detailed mathematical formulations can be found in texts such as [14], [49] and [54].

We start by formally introducing the 3D-Var method as this is the primary technique used for the work presented in this thesis. Later in the thesis we will take ideas from the Kalman and extended Kalman filters and so we also give details of the formulation and implementation of these schemes.

2.3.1 3D-Var and Optimal Interpolation

The 3D-Var method [13], [58], [74], is based on a maximum a posteriori estimate approach and derives the analysis by seeking a state that minimises a cost function measuring the misfit between the model state \mathbf{z}_k and the background state \mathbf{z}_k^b and the observations \mathbf{y}_k ,

$$J(\mathbf{z}_k) = \frac{1}{2}(\mathbf{z}_k - \mathbf{z}_k^b)^T \mathbf{B}_k^{-1}(\mathbf{z}_k - \mathbf{z}_k^b) + \frac{1}{2}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k))^T \mathbf{R}_k^{-1}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k)). \quad (2.6)$$

The matrices $\mathbf{B}_k \in \mathbb{R}^{m \times m}$ and $\mathbf{R}_k \in \mathbb{R}^{r_k \times r_k}$ are symmetric, positive definite covariance matrices associated with the background and observation errors.

We assume that the background and observation errors are unbiased and uncorrelated with Gaussian probability distribution functions. Their covariance matrices are then defined as

$$\mathbf{B}_k = E \left[\boldsymbol{\varepsilon}_{\mathbf{z}_k}^b \boldsymbol{\varepsilon}_{\mathbf{z}_k}^{b T} \right] \quad \text{and} \quad \mathbf{R}_k = E \left[\boldsymbol{\delta}_k \boldsymbol{\delta}_k^T \right], \quad (2.7)$$

where $E[\cdot]$ denotes expected value, $\boldsymbol{\varepsilon}_{\mathbf{z}_k}^b = \mathbf{z}_k^b - \mathbf{z}_k^t$ and $\boldsymbol{\delta}_k = \mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k^t)$.

These matrices represent the uncertainties of the background and observations and determine the relative weighting of \mathbf{z}^b and \mathbf{y}_k in the analysis. If it is assumed that the background errors are small relative to the observation errors then the analysis will be close to the background state. Conversely, if it is assumed that the background errors are relatively large the analysis will lie closer to the observations. If the error statistics are correctly specified then the analysis is a Bayesian maximum a posteriori estimate.

To determine the minimizing state at time t_k we use the gradient of the cost function with respect to \mathbf{z}_k . The analysis \mathbf{z}_k^a satisfies the equation

$$\nabla J(\mathbf{z}_k^a) = \mathbf{B}_k^{-1}(\mathbf{z}_k^a - \mathbf{z}_k^b) - \mathbf{H}_k^T \mathbf{R}_k^{-1}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k^a)) = 0, \quad (2.8)$$

where ∇J is the gradient of the cost function (2.6) with respect to \mathbf{z}_k , and the matrix $\mathbf{H}_k \in \mathbb{R}^{r_k \times m}$ represents the linearisation (or Jacobian) of the observation operator \mathbf{h}_k evaluated at the background state \mathbf{z}_k^b .

There are various ways of minimising the cost function, each of which leads to a different data assimilation scheme. When the observation operator, \mathbf{h}_k , is linear the minimum of (2.6) can be found exactly and the solution for the analysis can be written explicitly as

$$\mathbf{z}_k^a = \mathbf{z}_k^b + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}_k \mathbf{z}_k^b). \quad (2.9)$$

The operator $\mathbf{K}_k \in \mathbb{R}^{m \times r_k}$ is known as the *gain* matrix [74] and is given by

$$\mathbf{K}_k = \mathbf{B}_k \mathbf{H}_k^T (\mathbf{H}_k \mathbf{B}_k \mathbf{H}_k^T + \mathbf{R}_k)^{-1}. \quad (2.10)$$

Equation (2.9) is known as the best linear unbiased estimate or BLUE formula [54]. This equation forms the basis of one of the earliest and most simple data assimilation methods known as Optimal Interpolation (OI) or Statistical Interpolation [14], [27], [75]. The OI method uses (2.10) to calculate the gain matrix \mathbf{K} explicitly and solve (2.9) directly. An approximate version of the method was widely used for operational numerical weather forecasting throughout the 1970s and 80s before variational approaches became more common.

For problems where the dimension of the system and number of observations is large, such as global atmospheric and ocean forecasting, the matrices required in the calculation of \mathbf{K} are very large and therefore difficult and expensive to compute, store and invert. This means that the OI method is generally impractical for operational use in

environmental systems. It is, however, a valuable tool for low cost applications and offers a useful starting point for understanding data assimilation in simple systems. We adopt an OI approach in some of the simple numerical experiments we present later in the thesis. The 3D-Var method solves the nonlinear optimisation problem (2.6) numerically using a gradient descent algorithm to iterate to the minimising solution [28] and so is more efficient for large systems.

The prefix ‘3D’ refers to the fact that the 3D-Var method resolves the three spatial dimensions; it does not account for the time distribution of observations. Instead 3D-Var schemes are designed to produce an analysis at a single time point. Typically, in applications such as numerical weather prediction (NWP), observations are not taken simultaneously but are collected across a given time window. The convention in 3D-Var schemes is to assume that the state does not evolve significantly within this period and treats all observations as if they had been taken at the same time and assimilates them simultaneously. The analysis time is usually taken as the midpoint of the observation time window [59].

If a 3D-Var scheme is applied cyclically it can be regarded as a sequential data assimilation method. With sequential algorithms the model is evolved one step at a time, assimilating the observations in order. Each time a new set of observations becomes available they are combined with the current model forecast to produce an updated estimate of the current system state (the analysis). This analysis is then used to forecast the background for the next analysis time; the model is propagated forward to the time of the next observations, using the analysis as the initial state, and the assimilation process is repeated. We will use a sequential approach throughout this thesis. The steps of the forecast-assimilation cycle are summarised in algorithm 2.1.

Algorithm 2.1 The 3D-Var forecast-assimilation cycle

1. Starting from an initial estimate of the state \mathbf{z}_0^b at time t_0 , integrate the model (2.1) forward to the time t_k when a set of observations \mathbf{y}_k first become available.
2. Minimise the 3D-Var cost function (2.6) to obtain an updated estimate of the current model state \mathbf{z}_k^a .
3. Using the analysis \mathbf{z}_k^a as the initial state, evolve the model (2.1) forward to time

t_{k+1} . This then becomes the model background estimate \mathbf{z}_{k+1}^b at time t_{k+1} , i.e. $\mathbf{z}_{k+1}^b = \mathbf{f}(\mathbf{z}_k^a, \mathbf{p})$.

4. If no observational data are available at time t_{k+1} set $\mathbf{z}_{k+1}^a = \mathbf{z}_{k+1}^b$.
5. Repeat steps 3. and 4. until a new set of observations are made and then return to step 2.

The crucial difference between standard 3D-Var and four dimensional data assimilation schemes such as four dimensional variational data assimilation (4D-Var) and the Kalman filter is that the error covariance matrices are not evolved (implicitly or explicitly) by the 3D-Var algorithm. The background error covariance matrix has a fundamental impact on the quality of the analysis. Its prescription is therefore generally considered to be one of the most difficult and important parts in the construction of a data assimilation scheme [2], [3]. Rather than update \mathbf{B}_k at each new assimilation time, the 3D-Var method approximates this matrix once at the start of the assimilation and then holds it fixed throughout, as if the forecast errors were statistically stationary (i.e. $\mathbf{B}_k = \mathbf{B}$ for all k). It is therefore vital that it is given a good a priori specification. A large amount of research has been dedicated to developing methods for estimating the matrix \mathbf{B} . Useful reviews of current NWP techniques are given in [2], [3] and [24]. A full discussion of the structure and function of the background error covariance statistics in the context of variational data assimilation is given in [2]. Techniques for estimating \mathbf{B} are given in the companion paper [3]. The significance of the choice of background error covariance model in 3D-Var assimilation has prompted much of the work in this thesis. We discuss the role of the matrix \mathbf{B} in more detail in chapter 5.

2.3.2 The Kalman filter

The Kalman filter (KF) is a four dimensional sequential method. It was developed by Kalman [47] and Kalman and Bucy [48] and initially used in engineering applications. For a linear system, the KF algorithm produces an analysis that is (given the available observations and under certain statistical assumptions) statistically optimal in the sense that it is the minimum mean square error, or minimum variance, estimate [4], [45].

The main distinctions between the KF and 3D-Var are that the error covariances are evolved explicitly according to the model dynamics and the analysis is calculated directly. Instead of assuming that the background error covariance matrix is fixed, the KF forecasts \mathbf{B}_k forward, using knowledge of the quality of the current analysis to specify the covariances for the next assimilation step. This allows information from all previously assimilated observations to be taken into account, giving much greater observational impact.

In this section we present the KF algorithm for a discrete linear time-invariant model. We are assuming a perfect model (i.e. zero model error) but note that this is not a necessary assumption since the KF does allow for the inclusion of random model error (see for example [64]). The KF notation differs slightly from 3D-Var: the background state vector \mathbf{z}^b is replaced by the forecast vector \mathbf{z}^f to denote the fact that the background is now a forecast; the constant background error covariance matrix \mathbf{B} is replaced by the time-varying forecast error covariance matrix \mathbf{P}_k^f ; and we introduce a new matrix \mathbf{P}_k^a representing the analysis error covariance.

The Kalman filter predict and update equations

For a perfect, discrete linear time invariant dynamical system model

$$\mathbf{z}_{k+1} = \mathbf{M}_k \mathbf{z}_k \quad k = 0, 1, \dots \quad (2.11)$$

with observations linearly related to the state by the equations

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{z}_k + \boldsymbol{\delta}_k, \quad (2.12)$$

where $\mathbf{M}_k \in \mathbb{R}^{m \times m}$ is a non-singular matrix describing the dynamic evolution of the state from time t_k to time t_{k+1} and $\mathbf{H}_k \in \mathbb{R}^{r_k \times m}$ is a linear observation operator, the KF consists of the following steps:

Algorithm 2.2 The KF algorithm

1. State forecast

$$\mathbf{z}_{k+1}^f = \mathbf{M}_k \mathbf{z}_k^a \quad (2.13)$$

2. Error covariance forecast

$$\mathbf{P}_{k+1}^f = \mathbf{M}_k \mathbf{P}_k^a \mathbf{M}_k^T \quad (2.14)$$

3. Kalman gain computation

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^f \mathbf{H}_{k+1}^T (\mathbf{H}_{k+1} \mathbf{P}_{k+1}^f \mathbf{H}_{k+1}^T + \mathbf{R}_{k+1})^{-1} \quad (2.15)$$

4. Analysis

$$\mathbf{z}_{k+1}^a = \mathbf{z}_{k+1}^f + \mathbf{K}_{k+1} (\mathbf{y}_{k+1} - \mathbf{H}_{k+1} \mathbf{z}_{k+1}^f) \quad (2.16)$$

5. Analysis error covariance

$$\mathbf{P}_{k+1}^a = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1}) \mathbf{P}_{k+1}^f (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1})^T + \mathbf{K}_{k+1} \mathbf{R}_{k+1} \mathbf{K}_{k+1}^T. \quad (2.17)$$

If the Kalman gain \mathbf{K} (2.15) has been computed exactly the error covariance of the analysis (2.17) reduces to [26]

$$\mathbf{P}_{k+1}^a = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_{k+1}) \mathbf{P}_{k+1}^f. \quad (2.18)$$

The optimality of the KF solution depends on the assumptions underlying these equations being accurate. Note that the analysis equation (2.16) and the definition of \mathbf{K} (2.15) are the same as equations (2.9) and (2.10) for the BLUE with $\mathbf{B}_k = \mathbf{P}_k^f$.

2.3.3 The Extended Kalman Filter

The Kalman filter theory can be generalised for the case where the system model and/or observation operator are nonlinear by linearising around a background state. This gives the extended Kalman filter (EKF) [26], [45]. The steps of the EKF algorithm are the same as for the standard KF except:

The state forecast (cf. 2.13) is made using the full nonlinear model,

$$\mathbf{z}_{k+1}^f = \mathbf{f}(\mathbf{z}_k^a). \quad (2.19)$$

The matrix \mathbf{H}_k in equations (2.15) to (2.18) is replaced with the tangent linear of the nonlinear observation operator \mathbf{h}_k evaluated at the background state \mathbf{z}_k^f ,

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}(\mathbf{z}_k)}{\partial \mathbf{z}} \right|_{\mathbf{z}_k^f} \quad (2.20)$$

and the forecast error covariance matrix at time t_{k+1} , \mathbf{P}_{k+1}^f , is determined by propagating the analysis covariance forward in time from t_k using a linearisation of the nonlinear forecast model $\mathbf{f}(\mathbf{z})$,

$$\mathbf{P}_{k+1}^f = \mathbf{M}_k \mathbf{P}_k^a \mathbf{M}_k^T, \quad (2.21)$$

where

$$\mathbf{M}_k = \left. \frac{\partial \mathbf{f}(\mathbf{z}_k)}{\partial \mathbf{z}} \right|_{\mathbf{z}_k^a} \quad (2.22)$$

is the Jacobian of the nonlinear forecast model evaluated at the current analysis state \mathbf{z}_k^a .

Although the approximations made by the EKF make the optimisation problem easier to solve they do so at the expense of the optimality of the solution. Due to the linearisations of the observation and model operators, the optimal analysis property of the standard linear KF no longer holds and the actual analysis error may differ considerably from that implied by equation (2.17) [75].

2.3.4 Practical implementation

The KF and EKF methods are computationally much costlier than 3D-Var; the updating of the error covariance matrices requires the equivalent of $\mathcal{O}(m)$ model integrations, where m is dimension of the model state, plus adjoint and tangent linear models must be developed. The EKF works well for small, simple models but linearising complex models can be difficult, and if m is large the scheme becomes prohibitively expensive. Implementation of the full Kalman filter equations is therefore impractical for systems of high dimension. In practice the forecast error covariance matrix \mathbf{P}_k^f is kept constant or a much simpler updating is performed. However, the equations provide a useful starting point for the design and development of approximate algorithms, examples of which include the Ensemble Kalman filter (EnKF) [20], [40] and the reduced rank Kalman filter [23].

3D-Var is a robust and well established method that has many advantages, such as ease of implementation (no model adjoints required); computational robustness (given reasonably specified covariances) and computational efficiency. Consequently, the method

is a popular choice for state estimation in large problems. Although standard 3D-Var is designed to produce an analysis at a single time, by applying the method sequentially as part of an forecast-assimilation system we can utilise time series of observations. We use a sequential 3D-Var algorithm as a basis for the design of our joint state-parameter estimation scheme.

2.4 Summary

We have introduced the general dynamical model equations and described the general data assimilation problem for this system. We have given a brief overview of some of the different types of data assimilation before explaining the 3D-Var and Kalman filter data assimilation techniques in detail. These are the methods that we will utilize in subsequent chapters and so we also discussed some of the issues associated with their practical implementation. In the next chapter we discuss the problem of model parameter estimation and review examples of some of the different approaches to model calibration in coastal modelling applications.

Chapter 3

Parameter estimation

A mathematical model can only approximate reality; even the most advanced model will not be able to perfectly reproduce the true behaviour of a complex dynamical system. Parameters are intrinsic to environmental modelling. They are used as a way of representing processes that are not completely known or understood, or where limitations to computer power constrain the model resolution and therefore the level of detail that can be described. Sediment transport models, for example, are typically based on empirical formulae that use various parameterisations to characterise the physical properties of the sediment flux [81].

Poorly known parameters are a key source of uncertainty in model predictions [85]. Models often contain empirical or heuristic elements derived from practical experience rather than physical laws. A consequence of this is that model parameters often do not represent directly measurable quantities; their values therefore have to be assigned in other ways. A key question in model development is how to estimate these parameters a priori.

In section 3.1 we outline concepts essential to reliable parameter estimation. Then, in section 3.2, we describe some of the different methods used for the calibration of model parameters. For the purpose of reviewing the literature, we focus on coastal modelling applications. In section 3.3 we discuss a method for parallel state and parameter estimation known as dual estimation. In section 3.4 we introduce the technique of state augmentation and explain how data assimilation can be used to estimate uncertain

model parameters concurrently with the model state.

3.1 Ill-posedness, identifiability and uniqueness

The parameter estimation problem is often ill-posed. Problems occur due to instability of the parameter estimates to errors in the observational data and lack of uniqueness of the unknown parameter.

The stability of the parameter estimation solution is related to the continuity of dependence of the parameters on the observations. Instability arises when the values of the parameter estimates are sensitive to noise in the observed data meaning that small errors in the observations cause large errors in the estimated parameter value.

Non-uniqueness is linked to the concept of identifiability. Identifiability refers to whether there exists a unique solution to the parameter estimation problem. It is closely tied to the concept of observability [4], [12]; whether, for a given model structure, the available observations contain sufficient information for us to be able to determine the unknown parameters of interest. More precisely, the notion of identifiability addresses whether, given observations of the state variables and knowledge of the model dynamics, it is possible to obtain a unique deterministic set of model parameter values. If a parameter is not observable it will not be identifiable. A parameter estimation method can only be expected to work reliably when both these properties hold.

The review paper [22] gives a useful general introduction to the problem of parameter estimation in dynamical models. The authors show that in order to have a well-posed problem with a unique solution it is essential that the inverse problem is formulated in a consistent manner. This is done by penalising the errors in the observations, parameters, initial conditions and model equations and incorporating regularisation constraints to ensure parameter stability.

In [72] Navon gives descriptions of the issues of identifiability, ill-posedness and regularisation of the parameter estimation problem and stresses the importance of these concepts to robust parameter estimation through illustrative examples. He makes similar assertions to [22] and shows how regularisation of an ill-posed problem can result in

a well-posed problem with a physically meaningful solution.

Another important issue is parameter sensitivity [72], [93]; whether variations in model parameters result in detectable variations in model response. We cannot expect to be able to correct parameters that cause errors in the model solution that are on smaller scales than can be observed. It is therefore important to establish the sensitivity of the model state estimation errors to changes in the given parameters. In some instances, parameter observability and identifiability may be precluded by parameter interdependence or model structural weaknesses [93]. This can sometimes be resolved by an appropriate reparameterisation or restructuring of the model. Parameter sensitivity analysis can be a useful tool for identifying such problems. [72] also addresses this issue, advocating the use of adjoint relative sensitivity analysis as an efficient means for assessing the impact of a given parameter on the model forecasts, and thus as a guide for choosing the most important parameters for estimation.

3.2 Model calibration

Parameterisations are used across a wide range of applications. In most cases, parameter estimation is addressed as a separate issue to state estimation and model calibration is performed offline in a separate calculation. A useful introduction to model calibration is given in [93] in the context of hydrological modelling. The classic approach is manual calibration or ‘tuning’ of the model against observational data. Manual calibration can yield good results, but is a computationally and labour intensive and time-consuming process, requiring numerous repeated model runs. The results are also somewhat subjective with parameters often fitted via a visual comparison between the model output and observational data, without accounting for measurement error. This makes it difficult to assign any measure of uncertainty to the model predictions.

There are other approaches; the increase in computational capabilities in recent years has seen the development of many new, often complex, automated parameter optimisation algorithms. Generally, these methods involve the minimisation of an objective function that measures the misfit between modelled and observed quantities. The main distinctions between the different methods are how the minimum is located, how the

observational data are processed and the assumptions made about the statistics. A useful inter-comparison of several optimisation techniques for parameter estimation in terrestrial biogeochemical models is given in [101], including genetic algorithm, adjoint, Markov Chain Monte Carlo, Levenberg-Marquardt and the Kalman filter. In this study, it was found that, overall, each of the different methods were equally successful at estimating the optimum value of four parameters in a highly simplified terrestrial biosphere model. The choice of objective function appeared to have a more significant impact on the results than the choice of optimisation method.

3.2.1 Model calibration in coastal modelling

There are a variety of parameter estimation methods presented in the coastal modelling literature; the downhill simplex optimisation [38] and genetic algorithm [50], and hybrid genetic algorithm [84] techniques are based on determining a single ‘best’ parameter set through minimisation of a least squares cost function. Probabilistic approaches, such as classical Bayesian [107] and Bayesian Generalised Likelihood Uncertainty Estimation (GLUE) [85], [86] approaches are based on the principle that there is no single best parameter set and instead, the parameters are treated as probabilistic variables.

In [38], Hill et al. use a downhill simplex optimisation method to calibrate a 1D Lagrangian particle tracking model developed for modelling sediment resuspension rates in coastal waters. The simplex method is a local, direct search method. The termination point of a local search will depend on the point that it is started and so the method is not generally suited to practical problems for which there are likely to be multiple minima. The method was chosen for its robustness and to avoid the heavy computational burden associated with global search methods.

The approach was found to give significant results in all but one of the experiments presented; there was one test case for which no results were obtained due to the optimisation method failing to converge. In their discussion, the authors concede that further gain in performance might be made by employing a global optimisation technique.

A genetic algorithm is an example of a global search method. It is based on the ideas of evolution and natural selection and involves a population of individual parameter sets

that reproduce, mutate and die. In [50], the approach is used to calibrate an idealised model of bedform dynamics to bathymetric data. Here the optimal parameter combination is taken to be the individual set that gives the lowest root-mean-square difference between the model predicted and observed bathymetry. The method is found to improve predictions of sand bar migration when compared to experiments using theoretical parameter values. Only a short calibration period was required to get a long prediction window. The authors do not comment on the computational efficiency of their scheme. Because a genetic algorithm explores the entire parameter domain convergence to the global minimum can be slow. If the initial population size is large the number of iterations and cost function evaluations may require excessive computing time.

Generalised Likelihood Uncertainty Estimation (GLUE) [85], [86] is a Bayesian Monte Carlo simulation based technique that can be used to compute the posterior likelihood distributions of a model's parameters and to assess the predictive uncertainty associated with the parameter uncertainty. The idea that a single 'best' set of parameter values exists is cast aside. The method involves making multiple model runs with different parameter sets sampled from the feasible parameter space. A measure of goodness of fit is used to assign each parameter set a likelihood value; all those parameter sets that give model performance better than some user specified skill threshold are considered as acceptable whilst those below are discarded. These likelihoods are then used to derive uncertainty bounds for the model predictions. In [85], Ruessink applies the method to a model of nearshore bed evolution. It is found that acceptable results could be obtained with a relatively large range of parameter sets. The cause of this is identified as parameter interdependence and insensitivity caused by model structural errors. Ruessink argues that use of commonly favoured methods that search for a single set of parameter values would be inappropriate in this situation, and illustrates his assertion by applying a Levenberg-Marquardt local gradient based search algorithm to the same model. This problem is also discussed in [84].

The key advantages of the GLUE approach are that it allows for the assumption of non-Gaussian error statistics and provides an effective means for the quantitative assessment of the uncertainties inherent in numerical modelling. The main disadvantage is the inefficiency of the technique, arising from the repeated model calls to compute the likelihood value for each parameter set. Depending on the model complexity and parameter

sample size, this can be computationally very expensive and take a considerable amount of time. The experiments presented involved sampling 4100 parameter sets of which only around 1000 were found to meet the threshold for acceptance, therefore resulting in a large amount of spent computations. Another criticism is that the skill threshold is determined by the user. Its choice is therefore subjective and the subsequent uncertainty assessment will vary depending on the value chosen.

3.2.2 Discussion

There are examples demonstrating the effectiveness of each of the many different approaches to model parameter estimation. In terms of computational cost, set-up cost, adaptability, ease of implementation, etc. each methodology has different strengths and weaknesses. Often it will be the chosen application that will make certain methods more appropriate than others; the suitability of a particular approach will depend on factors such as model complexity, availability of observational data, computational resources and user expertise.

One distinction that is important in the context of this work is the way in which the observational data are used. Batch methods, such as genetic algorithm and Markov Chain Monte Carlo, process all observations at once. They essentially train the model against historical data and so become infeasible when there is a lack of sufficient observational information prior to the model forecast period. Sequential methods have the advantage that observations can be used as they arrive in real time. They allow the inclusion of new observations that become available after earlier observations have been assimilated thus enabling real-time state-parameter estimation. The inclusion of new observations when using batch methods would involve having to reprocess the earlier observations and therefore require the storage and processing of large amounts of data. Sequential methods also have the potential of being able to estimate parameters that are expected to vary over time. Sequential data assimilation techniques have the further advantage that they offer a framework for explicitly accounting for all sources of uncertainty.

3.3 Dual state-parameter estimation

Efforts in improving models tend to concentrate on either improving methods for parameter estimation (without explicitly accounting for uncertainty in the model state estimate) or improving methods for state estimation (estimating model variables whilst keeping the model parameters fixed).

A popular approach to state and parameter estimation is *dual* (as opposed to *joint*) state-parameter estimation. Here, the model state variables and model parameters are estimated in parallel using two separate calculations. In [104], Vrugt et al. present a hybrid sequential data assimilation and parameter optimisation framework for hydrologic modelling. The method, named SODA (simultaneous optimisation and data assimilation), uses an EnKF algorithm to recursively update the state variables in an inner loop whilst estimating the model parameters and their uncertainty in an outer global optimisation loop. The performance of the scheme is illustrated using the Lorenz 63 model and a classical, five parameter, conceptual watershed model (HYMOD). It performs well in both cases, producing improved estimates of the parameters and model prediction uncertainty. The state and parameter estimates produced by the SODA method for the HYMOD model experiments are shown to be consistently better than those obtained using a classic Bayesian model batch calibration method without state adjustments. The SODA method results in a considerable increase in computational burden but the authors argue that this is not an issue given current computing capacities. It could, however, limit the applicability of the method to more complex models. Notably, the authors mention that recursive parameter estimation via the state augmentation approach is being investigated as a possible method of improvement to the scheme.

Moradkhani et al. [70] use a dual EnKF approach in which the state and parameters are estimated by simultaneously running two mutually interactive filters. They use the same hydrologic model as [104] to demonstrate the algorithm, comparing results against three other common hydrologic model calibration algorithms. Their method performs equally well, with all parameter estimates converging smoothly. The main drawback of this scheme is that it involves running a separate filter for the parameters which then feeds into model state ensemble. The parameters are evolved artificially by assuming that they follow a random walk; this then requires implementation of a modified kernel

smoothing method to prevent the over dispersion of the parameter samples and the loss of information and introduces further computational complexity into the scheme.

Dual estimation methods offer an improvement on traditional model calibration methods in that they allow time series data to be utilised and enable information on the state adjustments to be fed back and incorporated into the parameter estimates. The downside is the inefficiency of the approach; arising from simultaneously running two algorithms in parallel rather than as a single process and having the potential to become prohibitively expensive in larger models. For the state-parameter interaction to be effective the feedback mechanism must be well defined; this may prove difficult in more complex models.

3.4 State augmentation

Data assimilation is most commonly used for state estimation. However, by employing the technique of *state augmentation* (also referred to as *joint estimation*), it is possible to use data assimilation to estimate uncertain model parameters concurrently with the model state. State augmentation is a conceptually straightforward technique that allows us to estimate and update uncertain model parameters jointly with the model state variables [45] as part of the assimilation process.

In theory, state augmentation can be applied with any of the standard data assimilation methods. The model state vector is augmented with a vector containing the parameters we wish to estimate, the equations governing the evolution of the model state are combined with the equations describing the evolution of these parameters and the chosen assimilation algorithm is simply applied to this new augmented system in the usual way. This enables us to estimate the model parameters and update the predicted model state simultaneously, rather than treating as two individual processes. This also means that observational data is used much more efficiently.

3.4.1 Applications of state augmentation

The state augmentation technique has previously been successfully employed in the context of model error or forecast bias estimation. Examples of application of the approach in simplified numerical models are given in [32], [33], [34], [62], [64]. In [17] the problem of forecast bias in atmospheric data assimilation is addressed using a sequential scheme with further discussion in [16]. In [5], [63] the technique is successfully used to estimate systematic errors in an operational equatorial ocean model.

The review article [72] discusses state augmentation for parameter estimation in relation to 4D-Var data assimilation techniques and surveys the literature relating to parameter estimation in meteorology and oceanography. State augmentation has also been used for parameter estimation with the Kalman filter; [100] investigated application of the technique with the EKF and EnKF for sequential assimilation of noisy pseudo-observations into a simplified two variable, four parameter terrestrial biosphere model. Both methods were successful at estimating the true parameters. Results were shown to be sensitive to choices of model and observation error covariance; the most stable parameter estimates were obtained when inflated estimates of the observation errors were used but this reduced the rate of convergence to the solution. The EnKF also showed a significant variation in results for different ensemble sizes. Even though the augmented state vector had only six components the EnKF required an ensemble of at least 50 members; this is similar to that used in much larger scale applications.

In [108] the state augmentation approach is combined with the maximum likelihood ensemble filter (MLEF) for model error estimation in the Korteweg-de Vries-Burgers (KdVB) model. The authors consider two approaches: parameter estimation and model bias estimation. Their results show that estimation of model error, either in the form of a parameter error or model bias, has a beneficial effect on the model; even in the cases where few observations and a small number of ensemble members are used, an improvement in filter performance was seen. Neglecting model error had an adverse effect on the MLEF, and in some cases resulted in filter divergence.

3.4.2 Mathematical formulation of the augmented system

In this work we consider application of the state augmentation technique to the problem of model parameter estimation using a sequential 3D-Var data assimilation scheme. In this section we reformulate the data assimilation problem for an augmented system. Although we present the technique within the framework of model parameter estimation, the procedure for model error or bias estimation is identical.

We use the vector $\mathbf{p} \in \mathbb{R}^q$ to represent the model parameters whose values are poorly known, where q is the number of parameters. We assume that these parameters are constants, that is, they are not altered by the forecast model from one time step to the next. The parameter estimates will only change when they are updated by the data assimilation at each new analysis time. The equation for the evolution of the parameters therefore takes the simple form

$$\mathbf{p}_{k+1} = \mathbf{p}_k \quad k = 0, 1, \dots \quad (3.1)$$

We define a new vector \mathbf{w} by extending the model state vector \mathbf{z} with the parameter vector \mathbf{p} , this gives us the *augmented state vector*

$$\mathbf{w} = \begin{pmatrix} \mathbf{z} \\ \mathbf{p} \end{pmatrix} \in \mathbb{R}^{m+q}. \quad (3.2)$$

Combining (3.1) with the model for the evolution of the state (2.1) we can write the equivalent augmented system model as

$$\mathbf{w}_{k+1} = \tilde{\mathbf{f}}(\mathbf{w}_k), \quad k = 0, 1, \dots \quad (3.3)$$

where

$$\tilde{\mathbf{f}}(\mathbf{w}_k) = \begin{pmatrix} \mathbf{f}(\mathbf{z}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix} \in \mathbb{R}^{m+q}. \quad (3.4)$$

In most cases the parameters will enter the model equations nonlinearly so that even if the original dynamical model is a linear function of the state variable the resulting augmented system model will be nonlinear. However, since the number of parameters is generally small relative to the dimension of the state vector and the dynamics of the

parameters are simple this does not have a significant impact on the computational cost of the data assimilation scheme.

The observations are related to the model states by the equations (2.2), as before. We can rewrite (2.2) in terms of the augmented state vector as

$$\mathbf{y}_k = \tilde{\mathbf{h}}_k(\mathbf{w}_k) + \boldsymbol{\delta}_k, \quad (3.5)$$

where $\tilde{\mathbf{h}}_k : \mathbb{R}^{m+q} \rightarrow \mathbb{R}^{r_k}$, and

$$\tilde{\mathbf{h}}(\mathbf{w}) = \tilde{\mathbf{h}} \begin{pmatrix} \mathbf{z} \\ \mathbf{p} \end{pmatrix} = \mathbf{h}(\mathbf{z}). \quad (3.6)$$

The analysis at time t_k , \mathbf{w}_k^a , is generated by combining the augmented model predictions with the observations using the chosen data assimilation algorithm. In addition to the updated state estimate, \mathbf{z}_k^a , the analysis will now also include updated estimates of the model parameters \mathbf{p}_k^a at each k .

The 3D-Var cost function for the augmented system takes the same form as (2.6) but is rewritten in terms of the new variables

$$J(\mathbf{w}_k) = \frac{1}{2}(\mathbf{w}_k - \mathbf{w}_k^b)^T \mathbf{B}_k^{-1}(\mathbf{w}_k - \mathbf{w}_k^b) + \frac{1}{2}(\mathbf{y}_k - \tilde{\mathbf{h}}_k(\mathbf{w}_k))^T \mathbf{R}_k^{-1}(\mathbf{y}_k - \tilde{\mathbf{h}}_k(\mathbf{w}_k)), \quad (3.7)$$

where

$$\mathbf{w}_k^b = \begin{pmatrix} \mathbf{z}_k^b \\ \mathbf{p}_k^b \end{pmatrix} \in \mathbb{R}^{m+q}. \quad (3.8)$$

is our background estimate of the augmented state \mathbf{w}_k at time t_k . Note that the initial background state at t_0 , \mathbf{w}_0^b , must include prior estimates of both the initial system state \mathbf{z}_o and parameters \mathbf{p}_0 . The analysis \mathbf{w}_k^a is then obtained by minimising (3.7) with respect to the augmented state vector \mathbf{w}_k .

The background error covariance matrix \mathbf{B}_k is redefined as

$$\mathbf{B} = E \left[\boldsymbol{\varepsilon}_{\mathbf{w}_k}^b (\boldsymbol{\varepsilon}_{\mathbf{w}_k}^b)^T \right] \in \mathbb{R}^{(m+q) \times (m+q)}, \quad (3.9)$$

where

$$\begin{aligned} \boldsymbol{\varepsilon}_{\mathbf{w}_k}^b &= \mathbf{w}_k^b - \mathbf{w}_k^t \\ &= \begin{pmatrix} \boldsymbol{\varepsilon}_{\mathbf{z}_k}^b \\ \boldsymbol{\varepsilon}_{\mathbf{p}_k}^b \end{pmatrix} \end{aligned} \quad (3.10)$$

with $\boldsymbol{\varepsilon}_{\mathbf{z}_k}^b = \mathbf{z}_k^b - \mathbf{z}_k^t$ and $\boldsymbol{\varepsilon}_{\mathbf{p}_k}^b = \mathbf{p}_k^b - \mathbf{p}_k^t$.

We stated in section 2.3.1 that the form of the matrix \mathbf{B}_k is very important. Constructing a realistic representation of the background error covariances is one of the key challenges of data assimilation. In basic state estimation the background error covariances govern how information is spread throughout the model domain, passing information from observed to unobserved regions and smoothing data if there is a mismatch between the resolution of the model and the density of the observations. Since it is not possible to observe the parameters themselves, the parameter updates will depend on the observations of the state variables. Practical implementation of the state augmentation approach therefore relies strongly on the relationships between the parameters and state components being well defined and assumes that we have sufficient knowledge to reliably describe them.

The augmented background error covariance matrix can be written as

$$\mathbf{B}_k = \begin{pmatrix} \mathbf{B}_{\mathbf{z}\mathbf{z}_k} & \mathbf{B}_{\mathbf{z}\mathbf{p}_k} \\ (\mathbf{B}_{\mathbf{z}\mathbf{p}_k})^T & \mathbf{B}_{\mathbf{p}\mathbf{p}_k} \end{pmatrix}. \quad (3.11)$$

Here $\mathbf{B}_{\mathbf{z}\mathbf{z}_k} \in \mathbb{R}^{m \times m}$ is the background error covariance matrix for the state vector \mathbf{z}_k at time t_k , $\mathbf{B}_{\mathbf{p}\mathbf{p}_k} \in \mathbb{R}^{q \times q}$ is the background error covariance matrix for the parameter vector \mathbf{p}_k and $\mathbf{B}_{\mathbf{z}\mathbf{p}_k} \in \mathbb{R}^{m \times q}$ is the covariance matrix for the cross correlations between the background errors in the state and parameters.

For joint state-parameter estimation, it is the cross covariances between the parameters and the state, given by the off diagonal blocks of the augmented background error covariance matrix (3.11), that transfer information from the observed variables to update the estimates of the unobserved parameters. This is most easily illustrated by considering the BLUE equation (2.9) for the augmented system [62]

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{y}_k - \tilde{\mathbf{H}}_k \mathbf{w}_k^b), \quad (3.12)$$

where

$$\tilde{\mathbf{h}}_k = \tilde{\mathbf{H}}_k \stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{H}_k & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{r_k \times (m+q)}. \quad (3.13)$$

By writing the gain matrix \mathbf{K}_k as

$$\begin{aligned}\mathbf{K}_k &= \mathbf{B}_k \tilde{\mathbf{H}}_k^T (\tilde{\mathbf{H}}_k \mathbf{B}_k \tilde{\mathbf{H}}_k^T + \mathbf{R}_k)^{-1} \\ &\stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{K}_{\mathbf{z}_k} \\ \mathbf{K}_{\mathbf{p}_k} \end{pmatrix},\end{aligned}\tag{3.14}$$

we can deconstruct (3.12) into state and parameter parts as follows

$$\mathbf{z}_k^a = \mathbf{z}_k^b + \mathbf{K}_{\mathbf{z}_k} (\mathbf{y}_k - \mathbf{H}_k \mathbf{z}_k^b)\tag{3.15}$$

$$\mathbf{p}_k^a = \mathbf{p}_k^b + \mathbf{K}_{\mathbf{p}_k} (\mathbf{y}_k - \mathbf{H}_k \mathbf{z}_k^b).\tag{3.16}$$

The gain matrices (3.14) are given by

$$\mathbf{K}_{\mathbf{z}_k} = \mathbf{B}_{\mathbf{z}\mathbf{z}_k} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{B}_{\mathbf{z}\mathbf{z}_k} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}\tag{3.17}$$

$$\mathbf{K}_{\mathbf{p}_k} = \mathbf{B}_{\mathbf{z}\mathbf{p}_k}^T \mathbf{H}_k^T (\mathbf{H}_k \mathbf{B}_{\mathbf{z}\mathbf{z}_k} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}.\tag{3.18}$$

Equation (3.15) is identical to the OI analysis equation (2.9) for state estimation. The equation for \mathbf{p}_k^a takes a similar form. The innovation vector $(\mathbf{y}_k - \mathbf{H}_k \mathbf{z}_k^b)$ is exactly the same in equations (3.15) and (3.16), as is the expression inside the inverse for the state and parameter gain matrices (3.17) and (3.18). The key difference is the presence of the cross covariance term $\mathbf{B}_{\mathbf{z}\mathbf{p}_k}$ in the gain matrix $\mathbf{K}_{\mathbf{p}_k}$. Both the state and parameters are updated according to the discrepancies between the observations and the model predicted state, the difference lies in exactly how this information is used. For the parameters, this is determined by the matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}}$. This is a crucial point; the state-parameter cross covariances play a vital role in the parameter updating, if they are inappropriately modelled the quality of the parameter estimates will be affected and this will in turn affect the predictive ability of the model. A good a priori specification of these covariances is fundamental to reliable joint state and parameter estimation. Since the correct error statistics of the system are generally unknown we have to approximate them in some manner.

In this thesis we explore different ways of approximating the matrix \mathbf{B}_k for concurrent state and parameter estimation using sequential data assimilation in simplified models. Ultimately, we hope to use the experience gained from work with these simple models to help guide the application of data assimilation based state and parameter estimation in operational forecasting systems.

3.5 Summary

In this chapter we have introduced the problem of model parameter estimation, highlighting the issues of ill-posedness, identifiability and uniqueness. We reviewed examples of some of the various approaches to model calibration in the context of coastal modelling. Finally, we introduced the technique of state augmentation and described how it can be used to estimate uncertain model parameters concurrently with the model state. In the next chapter we give an introduction to morphodynamic modelling and explain why joint state-parameter estimation would be a valuable aid to morphodynamic prediction.

Chapter 4

Morphodynamic modelling and data assimilation

This chapter is intended as a general introduction to coastal morphodynamic modelling. In section 4.1 we give the motivation for our choice of application; we explain why reliable, accurate morphodynamic models are important and highlight some of the difficulties associated with the development of an effective coastal morphodynamic model. In section 4.2 we present the sediment conservation equation and discuss parameterisation of the sediment transport flux. We summarise the various sources of observation data for morphodynamic data assimilation in section 4.3 before reviewing some recent applications of the approach to coastal modelling in section 4.4.

4.1 Morphodynamic modelling

Changes in coastal morphology can have wide reaching environmental and human social and economic impacts. Knowledge of how the coastline is evolving is essential for shoreline management and protection; informing planners and decision makers and ensuring coastal regions are monitored and developed appropriately. Unfortunately, knowledge of evolving near-shore bathymetry is limited; determination of the bathymetric changes occurring in the coastal zone is a key challenge in coastal modelling. State of the art computational models are becoming increasingly sophisticated (e.g. [51]) but in prac-

tice these models suffer from uncertainty in their initial conditions and parameters. Improved accuracy of morphodynamic models would allow better prediction of future bathymetry, enabling the development of more effective forecasting systems and thus allowing us to better prepare for, or even prevent, the effects of morphodynamic change.

An increase in extreme and hazardous weather events in recent years, has led to growing concern over the effects of climate change. Expected sea level rise combined with an increase in the frequency and intensity of storms has profound implications for coastal flooding and further highlights the need for better knowledge and understanding of how the morphology of the coast is changing [73]; bottom topography is a large source of uncertainty in coastal inundation modelling and can strongly influence the quality of model predictions [10], [37]. Accurate bathymetry immediately prior to a storm event would be a valuable aid to the advancement of inundation forecasting and the mitigation of flood hazard.

Coastal morphodynamic modelling is challenging; an effective coastal morphodynamic model must be able to represent the continual interaction between water flow and bathymetry in the coastal zone. Modelling is difficult because longer term morphological changes are driven by shorter term processes such as waves, tides and river outflows [68]. The bathymetry of the coastline changes as sediment is eroded, transported and deposited by water action. The change in bathymetry alters the water flow, which further changes the bathymetry, which in turn alters the motion of the water, and so on [68], [76]. Ideally, these processes should be treated simultaneously, but it is common to split models into hydrodynamic and bathymetric components; the water flow is updated while the bathymetry is held constant, and vice versa [15], [83]. In this thesis we focus on modelling the bathymetric changes in a simplified one dimensional system. A derivation of the full equations governing morphodynamic change can be found for example in [42], [103], [94].

4.2 Modelling the sediment transport flux

Changes in bathymetry due to flow induced sediment transport processes, can be described using the sediment conservation equation [94]. In one dimension this is written

as

$$\frac{\partial z}{\partial t} = - \left(\frac{1}{1 - \epsilon} \right) \frac{\partial q}{\partial x}, \quad (4.1)$$

where $z(x, t)$ is the bathymetry or bed height, t is time, q is the total (suspended and bedload) sediment transport rate in the x direction and ϵ is the sediment porosity.

The sediment porosity ϵ is a non-dimensional value, expressed as a fraction between 0 and 1, that depends on the degree of sediment sorting and compaction. Its value can be obtained using the measurement techniques described in [94]. In this thesis, we set $\epsilon = 0.4$ which corresponds to a natural sand bed with average sorting and packing. This is the recommended default value to be used in the absence of any other information [94].

The transport rate q is a complex function of the water properties such as currents, waves and water depth plus characteristics of the sediment such as density and grain size [55]. In order to solve (4.1) q needs to be estimated.

We wish to construct a model that satisfies the physical properties of the system as closely as possible. Unfortunately, modelling is limited by our inability to fully discern the various processes governing sediment transport. The lack of detail in our knowledge means that parameterisations must be used.

There is no universally agreed formulation for q ; numerous different formulae have been proposed, many of which are presented in [94] and [103]. These are typically based on a mixture of theory and empirical results and are often only valid under certain assumptions, such as, the sediment grain size lies within a given range. The choice of formula is generally dependent on the particular situation being modelled, with parameter values being calibrated to fit the physical characteristics of the study site. The level of computational resources available is also a relevant factor. In [81] the strengths, weaknesses and applicability of four popular sediment transport formulae are assessed and references to other comparison studies are also given. The authors state that all four formulae have limitations. However, it is found that the accuracy of the sediment flux evaluation is dominated by errors in the values assigned to physical input parameters, rather than by the limitations of the formulae themselves. The key physical properties being the current velocity and the sediment median grain size. Despite often being regarded as

one of the best formulae available, the combined bedload and suspended-load transport formulae of Van Rijn [103] is found to be the most sensitive to errors in basic physical properties.

For the work presented in this thesis, we use one of the most basic sediment transport flux formulae. Here, the various processes driving sediment transport are combined in a single equation. Originally proposed by [30], the flux is assumed to follow the simple power law equation,

$$q = Au|u|^{n-1}, \quad (4.2)$$

where $u = u(x, t)$ is the depth-averaged current in the x direction and A and n are parameters whose values need to be set. The parameter A is a dimensional constant whose value depends on various properties of the sediment and water, such as flow depth and velocity range, sediment grain size and kinematic viscosity. [103] gives a formula that can be used to obtain an approximation to A for a given set of sediment and water properties. The derivation of the parameter n is less clear. It is usually set by fitting to field data and generally takes a value in the range $1 \leq n \leq 4$.

Equation (4.2) provides a estimate of the combined bedload and suspended-load sediment transport rate. It has been chosen for its relative simplicity and is more than adequate for the purposes of this research. This allows us to focus on the application of data assimilation techniques and avoids unnecessary model complexity.

In order to generate reliable forecasts of long-term morphodynamic evolution we need to have good estimates of both the model parameters and the current bathymetry. Parameter uncertainty is a major source of error in morphodynamic modelling; even if the initial bathymetry is well known, errors in the parameters will affect the accuracy of the sediment transport flux calculation, leading to the growth of model error and in turn the accuracy of the predicted bed level changes [85]. Despite its disadvantages, manual calibration is still the most popular model calibration method within the coastal modelling community [86].

4.3 Observation sources

Data assimilation requires observational data. For the work in this thesis we use pseudo observations. However, in principle there are a number of potential sources of observational data that could be used for operational coastal morphodynamic prediction. For completeness we give a brief summary of some of these sources. More detailed descriptions can be found in the references given.

An important benefit of data assimilation is that observations do not need to provide complete coverage of the model domain. Observations of bathymetry are available from a variety of sources. Mason et al. [67] provide a comprehensive comparison of several mapping techniques including ground surveys e.g. GPS (Global Positioning System) and beach transects, aerial surveys e.g. airborne LiDAR (Light Detection and Ranging), satellite interferometry e.g. SAR (Synthetic Aperture Radar) imaging, and the waterline method (see also [65] and [87]). Other examples include ship based echo soundings (swath bathymetry), X-band radar [6] and Argus video systems [39], [102].

Obviously, not all types of observations will be available in all cases. The suitability of an observation method for a given application will depend on a number of factors such as the geographical properties of the site under study, whether we are interested in modelling short term or long term behaviour and the scale of changes being predicted, i.e. whether the scales observed are compatible with the desired model resolution. Cost is also often a necessary consideration.

For example, ground surveys (e.g. [29]) can provide observations to a high level of accuracy but are not really suitable for large regions and, depending on the field site, may not even be logistically possible. Airborne surveys can cover large areas in short times and areas inaccessible from the ground but generally require reasonably good weather conditions.

LiDAR (e.g. [36]) is a newer airborne mapping technology that can provide rapid, accurate elevation data with a high coverage rate. The method uses laser pulses to measure the distance between the survey aircraft and the ground surface. It was once considered expensive but the technology is now becoming much more affordable and surveys are being carried out on a far more frequent basis.

For areas inaccessible to both humans and aircraft, the waterline method is a useful technique [65], [67]. A waterline is essentially a contour of bathymetry representing the land-sea boundary at a particular instant in time. They are derived using a remotely sensed image of the region, such as a SAR satellite image; these are radar images that simply show which parts of the coastal zone are under water. The waterline is extracted using image processing techniques and assigned a height using a hydrodynamic model of the region. Although this is manually intensive work it is worthwhile because SAR images provide a relatively cheap source of partial bathymetry, thus making them ideal for data assimilation [87]. The main disadvantage of this method is that it is only suitable for inter-tidal areas.

In order to specify the observation error variance and covariances, data assimilation requires information about the accuracy of the observations being assimilated. Knowledge of the various possible sources and estimated size of the errors associated with an observation type is therefore important. Direct measurements of bathymetry are obviously much easier to use than indirect observations which require the design of a more complex (nonlinear) observation operator and may introduce further errors. The use of remotely sensed observations such as SAR satellite images and radar images will also involve an additional pre-processing stage.

4.4 Morphodynamics and data assimilation

While data assimilation has been used for state estimation in atmospheric and oceanic prediction for some years, it has rarely been employed for morphodynamic modelling, despite the availability of a variety of suitable observations. In a precursor to the current work, Scott and Mason [87] explored the use of data assimilation for state estimation in estuarine morphodynamic modelling using Morecambe Bay as a study site. A 2D horizontal (2DH) decoupled morphodynamic model of the bay was enhanced by integrating waterline observations derived from SAR satellite images [66] using a simple OI assimilation scheme. Despite the known deficiencies of the OI algorithm (see e.g. [56]), the method was shown to improve the ability of the model to predict large scale changes in bathymetry over a three year period. In an unrelated study, [102] used a least squares estimator to assimilate observations from multiple, remotely sensed sources

(video and radar) into the Delft 3D modelling system. This system did not take account of spatial correlations between model variables and thus only updated model variables where there were co-located observations. Nevertheless, the system showed good skill in estimating the nearshore subtidal bathymetry when applied to two data-rich test sites at Duck, NC, USA and Egmond, The Netherlands.

In this thesis, we address state and parameter estimation as a single issue. To the best of our knowledge data assimilation has not been used for joint state-parameter estimation in morphodynamic modelling before. The hypothesis is that this approach will enable more efficient state and parameter estimation, thereby saving on calibration time, making better use of the available observational data and giving more accurate forecasts.

4.5 Summary

We have given an introduction to morphodynamic modelling and discussed some of the difficulties associated with the development of an effective coastal morphodynamic model. We described some the different observation sources available for morphodynamic data assimilation and reviewed examples of previous applications of data assimilation to coastal modelling. In the next chapter we begin to develop our joint state-parameter estimation system.

Chapter 5

System development

In this chapter we start to build our joint state-parameter estimation system. We use a series of experiments to help demonstrate the theory and highlight some of the issues associated with practical implementation of the 3D-Var and state augmentation techniques.

We begin by using a simple 1D linear framework. The model we employ is idealised and has been chosen as a first-step model that allows ideas to be developed, tested and understood without the obfuscating features of a more complex system.

We start this chapter by describing the model and its discretisation. In section 5.2 we introduce and test the algorithm used to minimise the 3D-Var cost function. In section 5.3 we discuss the roles of the background and observation error covariance matrices \mathbf{B} and \mathbf{R} . We pay particular attention to the background error correlations and the way in which they govern the spatial spreading and smoothing of information from the observations. We illustrate the importance of an appropriate choice of background error covariance model via a series of simple state estimation assimilation experiments using four alternative forms for the state background error covariance matrix \mathbf{B}_{zz} . In section 5.4 we move on to the augmented data assimilation problem and consider ways of defining the state-parameter cross covariance matrix \mathbf{B}_{zp} required for joint state-parameter estimation. Sections 5.5 and 5.6 describe experiments using two different approaches. We note that some of the work in sections 5.4 to 5.6 has been previously published in the journal article [90].

5.1 Linear advection

As we discussed in chapter 4, section 4.1, the physics underlying morphodynamic change are extremely nonlinear and difficult to model. Here, rather than focus on the development of an accurate model, we wish simply to demonstrate the data assimilation technique and assess the potential of the state augmentation approach for concurrent state-parameter estimation. For this purpose, we consider a highly simplified scenario in which the morphodynamic evolution of our system is described by the 1D linear advection equation [53], [71], [88].

Re-writing the sediment conservation equation (4.1) in quasi-linear form gives

$$\frac{\partial z}{\partial t} + c(z, q) \frac{\partial z}{\partial x} = 0, \quad (5.1)$$

where

$$c(z, q) = \left(\frac{1}{1 - \epsilon} \right) \frac{\partial q}{\partial z}. \quad (5.2)$$

The coefficient $c(z, q)$ is the advection velocity or bed celerity. It is a nonlinear function, depending on the bathymetry z both directly and through the sediment transport rate q .

To simplify, we assume that the advection velocity is constant and replace the coefficient $c(z, q)$ with a single constant parameter c , equation (4.1) then reduces to

$$\frac{\partial z}{\partial t} + c \frac{\partial z}{\partial x} = 0. \quad (5.3)$$

The model (5.3) has the benefit of an analytical solution; this can be used both to provide observations for the assimilation and also to verify the results of our scheme.

For known, non-zero, constant c and given initial data

$$z(x, 0) = z_0(x), \quad -\infty < x < \infty, \quad (5.4)$$

the solution at time $t \geq 0$ is simply [53]

$$z(x, t) = z_0(x - ct). \quad (5.5)$$

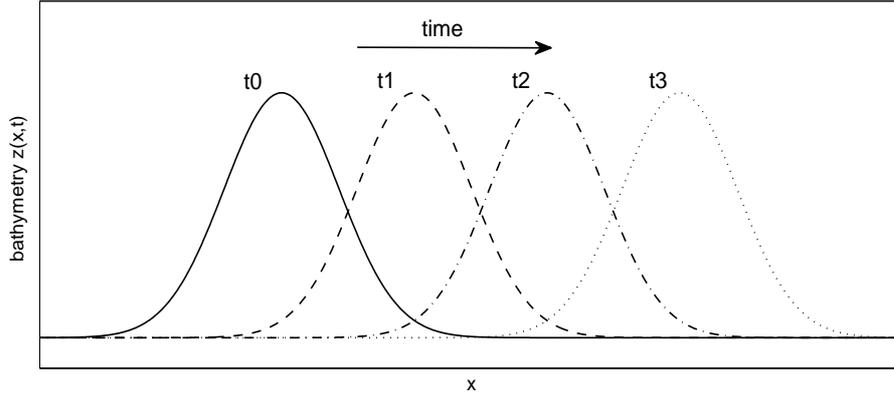


Figure 5.1: Schematic: solutions to the linear advection equation (5.3) for Gaussian initial data at times t_0 (solid line), t_1 (dashed line), t_2 (dot-dash line) and t_3 (dotted line), $t_0 < t_1 < t_2 < t_3$.

For the purpose of the experiments in this chapter, we assume that the true initial bathymetry can be described by the Gaussian function

$$z_0(x) = \alpha e^{-(x-\beta)^2/2\gamma^2}. \quad (5.6)$$

This gives a smooth, isolated, bell-shaped bedform where the constants α , β and γ determine the height, initial position and width respectively.

The solution (5.5) has the property that it preserves its initial shape $z_0(x)$. As figure 5.1 illustrates, as time evolves, the initial data propagates undistorted at constant speed c to the right (if $c > 0$).

5.1.1 Discretisation

For our assimilation scheme we require $z(x, t)$ at discrete points (x_j, t_k) . We wish to solve (5.3) on a uniform mesh with spatial grid spacing Δx and model time step Δt . Let $z(x_j, t_k)$ denote the value of the solution at the point (x_j, t_k) , where $x_j = j\Delta x$ and $t_k = k\Delta t$. A common approach to the numerical modelling of advection is the semi-Lagrangian technique [97].

Consider the total (or *Lagrangian*) time derivative of z ,

$$\frac{dz}{dt} = \frac{\partial z}{\partial t} + \frac{dx}{dt} \frac{\partial z}{\partial x} = 0. \quad (5.7)$$

From (5.3) we have

$$\frac{dx}{dt} = c. \quad (5.8)$$

In other words, the bed height $z(x, t)$ remains constant along the trajectory (5.8).

We suppose that we know $z(x, t)$ at all points on the model grid at time t_k and that we want to find values on the same regular grid at time t_{k+1} . The general principle behind the semi-Lagrangian method is that at each new time step t_{k+1} we can trace back the trajectory of the particle arriving at the grid point x_j to its position at the previous time t_k and hence find the value $z(x_j, t_{k+1})$. We refer to the point x_j at time level t_{k+1} as the arrival point. The position of the particle at the previous step t_k is known as the departure point and is denoted x_d .

We approximate (5.7) with the discretization

$$\frac{z(x_j, t_{k+1}) - z(x_d, t_k)}{\Delta t} = 0 \quad (5.9)$$

Hence the value of z at the arrival point x_j at time t_{k+1} is equal to the value of z at the departure point x_d at time t_k , i.e.

$$z(x_j, t_{k+1}) = z(x_d, t_k). \quad (5.10)$$

Generally, the point x_d will not lie on the model grid. However we can easily obtain the value $z(x_d, t_k)$ by interpolation from neighbouring grid points.

The departure points x_d are calculated as

$$x_d = x_j - \alpha_j, \quad (5.11)$$

where α_j is the displacement of the particle in the x direction between t_k and t_{k+1} .

In the case of linear advection, the advection speed c is constant and we can compute α_j exactly

$$\alpha_j = \alpha = c\Delta t, \quad \text{for all } j \quad (5.12)$$

giving

$$x_d = x_j - c\Delta t. \quad (5.13)$$

We summarise the steps of the algorithm as follows:

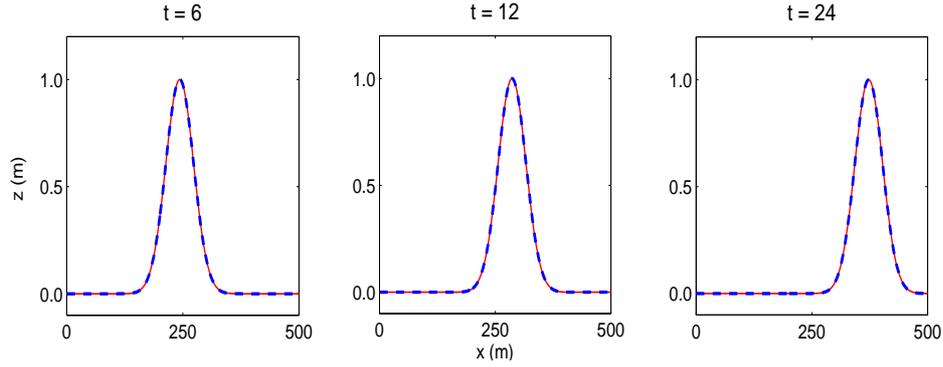


Figure 5.2: Comparison of analytic and numerical solutions to the linear advection equation (5.3) with Gaussian initial data at times $t = 6, 12$ & 24 hours. The solid red line represents the analytic solution, the dashed blue line is the solution computed using a semi-Lagrangian scheme with grid spacing $\Delta x = 1.0\text{m}$ and time step $\Delta t = 1$ hour.

Algorithm 5.1 The semi-Lagrangian algorithm

- (i) Calculate the departure points x_d using (5.13).
- (ii) Given $z(x_j, t_k)$, evaluate z at departure points x_d at time t_k using an interpolation formula.
- (iii) Evaluate z at arrival points x_j at time t_{k+1} using equation (5.10).

The model was validated by comparing its output with the analytic solution (5.5). Figure 5.2 compares the numerical (dashed blue line) and analytic (solid red line) solutions at times $t = 6, 12$ and 24 hours for Gaussian initial data with $c = 0.002 \text{ ms}^{-1}$ and spatial resolution $\Delta x = 1.0\text{m}$. In this example, the semi-Lagrangian algorithm is run with time step $\Delta t = 1$ hour. It is almost impossible to distinguish between the two solutions. The advantage of semi-Lagrangian methods is that they are unconditionally stable; this allows us to use larger time steps and achieve a higher order of accuracy at a low computational cost. In chapter 8 we use the same approach to solve the nonlinear sediment conservation equation (5.1). Details of the method are given in section 8.2.

5.1.2 Parameter model

We assume that advection velocity is constant over time, but that its true value is not known with certainty and is not directly measurable. The evolution model for c is given

by

$$\frac{dc}{dt} = 0. \quad (5.14)$$

We can write (5.14) in discrete form as

$$c_{k+1} = c_k, \quad k = 0, 1, \dots \quad (5.15)$$

Equation (5.15) together with the model equation (5.9) constitutes our augmented state system model (3.3).

5.2 The minimisation algorithm

As described in section 2.3.1, the 3D-Var analysis at time t_k is generated by minimising the cost function (2.6)

$$J(\mathbf{z}) = \frac{1}{2}(\mathbf{z}_k - \mathbf{z}_k^b)^T \mathbf{B}_k^{-1}(\mathbf{z}_k - \mathbf{z}_k^b) + \frac{1}{2}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k))^T \mathbf{R}_k^{-1}(\mathbf{y}_k - \mathbf{h}_k(\mathbf{z}_k)). \quad (5.16)$$

Implementation of the 3D-Var method requires a suitable numerical descent (minimisation) algorithm. For the work described in this this thesis we use a quasi-Newton method. We employ the Matlab function `quasi_newton.m` written by Hans Bruun Nielsen of DTU (Technical University of Denmark). A full description of the quasi-Newton algorithm can be found in the documentation [25] along with references and details of other iterative methods including steepest descent, conjugate gradient, and damped Newton.

The quasi-Newton algorithm takes value of the cost function J and its gradient ∇J as input arguments and uses the current background state as a starting point for the iteration. The inverse Hessian $(J'')^{-1}$ is approximated using the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula [77] and this is combined with a soft line search [78]. The minimisation is terminated by one of: a sufficiently small gradient $\|\mathbf{f}'(\mathbf{x}_k)\|_\infty \leq \varepsilon_1$, a sufficiently small step length $\|\mathbf{x}_k - \mathbf{x}_{k-1}\|_2 \leq \varepsilon_2(\varepsilon_2 + \|\mathbf{x}_k\|_2)$, the number of iterations exceeding a user specified maximum k_{\max} or a zero step. For the work presented in this thesis, the tolerances on these stopping criteria are set at $\varepsilon_1 = 10^{-12}$, $\varepsilon_2 = 10^{-12}$ and $k_{\max} = 500$.

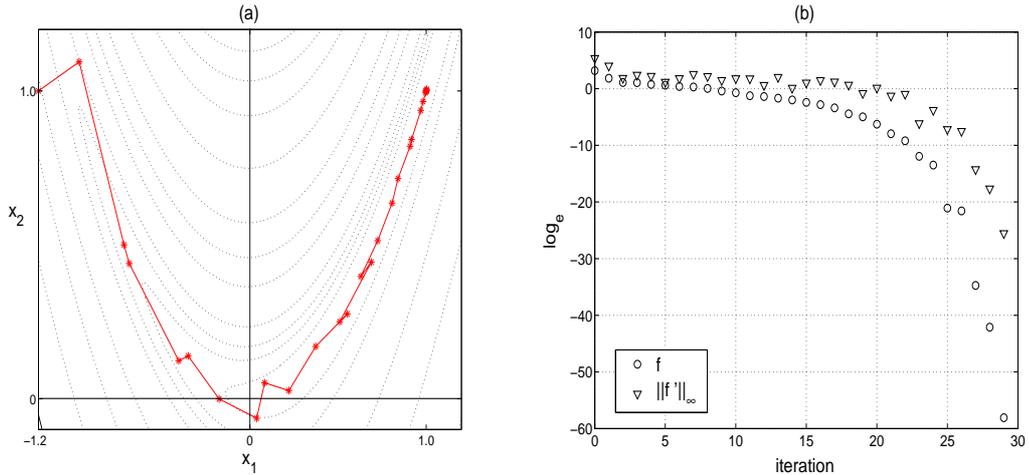


Figure 5.3: The quasi-Newton method applied to Rosenbrock’s function: (a) contours (dotted black lines) and iterates \mathbf{x}_k (solid red line), (b) values of $f(\mathbf{x}_k)$ (circles) and $\|\mathbf{f}'(\mathbf{x}_k)\|_\infty$ (triangles). Note the logarithmic scale.

5.2.1 Verification

The algorithm was verified using several different test functions with known minimisers, examples of which are given in [25]. Figure 5.3 gives the iterates and the values of $f(\mathbf{x}_k)$ and $\|\mathbf{f}'(\mathbf{x}_k)\|_\infty$ when applying the method to Rosenbrock’s function

$$f(\mathbf{x}) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2. \quad (5.17)$$

with initial guess $\mathbf{x}_0 = [-1.2, 1]^T$. This is a standard nonlinear test function used in optimisation theory. The function (5.17) has a unique global minimum at $\mathbf{x}^* = [1, 1]^T$ with $f(\mathbf{x}^*) = 0$.

5.3 Error covariances

Error covariances play an important role in variational data assimilation. Before we can implement our 3D-Var algorithm we need to specify the background and observation error covariance matrices \mathbf{B} and \mathbf{R} .

We are assuming that our model structure is perfect, i.e. with the correct choice of parameter the model equations provide an exact representation of the dynamical system. Obviously, this assumption is unrealistic. In practice, it is impossible to describe the true system behaviour completely, and model predictions will also contain errors as a result

of uncertain parameters and inaccurate initial and boundary conditions. In addition, the observations we wish to assimilate are likely to incorporate some kind of error, however small. Our assimilation scheme needs to take account of the errors that arise as a result of these imperfections. The precision of the analysis is dependent on the precision of the background and observations: correct specification of the error covariance matrices \mathbf{B} and \mathbf{R} is therefore crucial to the success of the scheme.

Since prescription of the background error covariance matrix \mathbf{B} is generally considered to be one of the most difficult parts in the construction of a data assimilation scheme, the majority of this section is devoted to discussion of this matrix. We begin, however, with a brief outline of the role of the observation error covariance matrix \mathbf{R} .

5.3.1 Observation error covariance

The observation error covariance matrix \mathbf{R} gives a statistical description of the errors in \mathbf{y} . Observation errors originate from instrumental error, errors in the forward model \mathbf{h} and representativeness errors (observing scales that cannot be represented in the model [9]). Generally, it is reasonable to assume that errors in measurements taken at different locations are independent. In this case the observation error correlations are zero and the matrix \mathbf{R} is a diagonal with error variance σ_o^2

$$\mathbf{R}_k = \sigma_o^2 \mathbf{I}_k, \quad \mathbf{I} \in \mathbb{R}^{r_k \times r_k}. \quad (5.18)$$

5.3.2 State background error covariance

Specification of the background error covariance matrix is one of the key parts of the assimilation problem. Background errors arise from errors in both the initial conditions and model errors. Since, by the nature of the problem, these errors are not known exactly, they have to be approximated in some manner.

For state estimation, the state background error covariance matrix (which we will denote $\mathbf{B}_{\mathbf{z}\mathbf{z}} = \{b_{ij}\}$) represents the estimation errors of the background state vector \mathbf{z}^b . Element b_{ij} of $\mathbf{B}_{\mathbf{z}\mathbf{z}}$ defines the covariance between components i and j of the state background

state error vector $\boldsymbol{\varepsilon}_{\mathbf{z}}^b$, where $\boldsymbol{\varepsilon}_{\mathbf{z}}^b = \mathbf{z}^b - \mathbf{z}^t$ is the difference between the background state vector \mathbf{z}^b and the true state vector \mathbf{z}^t .

The correlations in $\mathbf{B}_{\mathbf{zz}}$ govern the spreading and smoothing of the observational information, determining how an observation at one point influences the analysis at nearby points [9]. They are therefore fundamental in determining the nature of the resulting analysis. In noisy, observation dense regions, the background error correlations are used to ensure that the analysis is smooth and contains scales that are statistically compatible with the smoothness properties of the physical fields. In data sparse regions, where there are few observations, correlations are needed to spread the information contained in the available observations to the surrounding domain [2], [14]. If the background error correlations are poorly specified the quality of the analysis can be considerably reduced.

Formulation of the background error covariance can be made considerably easier by specifying the error correlations as analytic functions. A number of correlation models have been proposed (see [14] for further discussion on this). An approach commonly used by the NWP community is the National Meteorological Center method (NMC) [79], which uses the difference between forecasts that verify at the same time. The literature gives various other methods, including using innovation (observation minus background) statistics and studying differences in background fields using ensemble techniques. The seminar paper by Fisher [24] and more recent review papers by Bannister [2], [3] provide a useful overview of current NWP techniques. A standard approach used in state estimation is to assume that the background error covariances are homogeneous and isotropic. The matrix $\mathbf{B}_{\mathbf{zz}}$ is then equal to the product of the estimated error variance and a correlation matrix defined using a pre-specified correlation function. Although this method is somewhat crude it makes the data assimilation problem far more tractable.

5.3.3 Experiments

We illustrate the effect the choice of state background error covariance model can have on the analysis by comparing the results produced by four different $\mathbf{B}_{\mathbf{zz}}$ matrices in a simple state estimation experiment.

We start with a simple diagonal matrix

$$\mathbf{B}_{\mathbf{z}\mathbf{z}} = \sigma_b^2 \mathbf{I} \quad \mathbf{I} \in \mathbb{R}^{m \times m}, \quad (5.19)$$

where σ_b^2 is the background error variance.

Next we add entries above and below the main diagonal by setting

$$b_{i-1,i} = b_{i,i-1} = \frac{\sigma_b^2}{2}, \quad i = 2, \dots, m. \quad (5.20)$$

This gives the tri-diagonal matrix

$$\mathbf{B}_{\mathbf{z}\mathbf{z}} = \begin{pmatrix} \sigma_b^2 & \frac{\sigma_b^2}{2} & 0 & \dots & \dots & 0 \\ \frac{\sigma_b^2}{2} & \sigma_b^2 & \frac{\sigma_b^2}{2} & 0 & \dots & \vdots \\ 0 & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & \ddots \\ 0 & \dots & \dots & 0 & \frac{\sigma_b^2}{2} & \sigma_b^2 \end{pmatrix}. \quad (5.21)$$

Finally, we consider two examples of state background error covariance matrices that have full off diagonal entries. We use two analytical models of the form $\mathbf{B}_{\mathbf{z}\mathbf{z}} = \sigma_b^2 \boldsymbol{\rho}$ where $\boldsymbol{\rho} = \{\rho_{ij}\}$ is a correlation matrix.

The first has elements given by the Gaussian exponential function [14]

$$\rho_{ij} = e^{-r_{ij}^2/2L^2}, \quad i, j = 1, \dots, m. \quad (5.22)$$

The second is the Markov matrix [82]

$$\rho_{ij} = e^{-|r_{ij}|/L} \quad i, j = 1, \dots, m. \quad (5.23)$$

The element ρ_{ij} defines the correlation between components i and j of the state background error vector $\boldsymbol{\varepsilon}_{\mathbf{z}}^b = \mathbf{z}^b - \mathbf{z}^t$. Here $r_{ij} = (x_i - x_j)$ is the distance between the grid points x_i and x_j , and L is the background correlation length scale.

These models give a full symmetric error covariance matrix with variance σ_b^2 on the diagonal and non-zero off-diagonal elements. The matrices (5.22) and (5.23) assume that the correlation between the background errors at two grid points depends only on the spatial distance between them and that correlations decrease with separation distance.

5.3.4 Results

At this stage we do not consider the time evolution of the state; we perform the assimilation at a single time point with each of the different \mathbf{B}_{zz} matrices (5.19) - (5.23).

The true bathymetry is given by the Gaussian function

$$z^t(x) = 0.5e^{-50(x-0.4)^2}. \quad (5.24)$$

The background guess is also taken to be Gaussian but with slightly different scaling factors

$$z^b(x) = 0.25e^{-10(x-0.45)^2}. \quad (5.25)$$

We use a uniform model grid with observations taken from the truth at equally spaced intervals. These observations are assumed to be perfect and without any added noise.

The background and observation error variances σ_b^2 and σ_o^2 should reflect our uncertainty in the background and the observations. Since the observations are taken from the truth and assumed to be perfect we weight in their favour, setting $\sigma_b^2 = 1.0$ and $\sigma_o^2 = 0.1$.

Figure 5.4 shows the analyses produced for each of the background covariance matrices on the domain $x \in [0, 1]$ with grid spacing $\Delta x = 0.02$, and observations every $10\Delta x$, so that the dimension of the state vector $m = 51$ and $r = 6$. The red dot-dash line represents the true bathymetry \mathbf{z}^t . The observations \mathbf{y} are given by circles, the background \mathbf{z}^b by the blue dashed line and the analysis \mathbf{z}^a by the solid green line. Alternative observation combinations were also investigated but for brevity we do not show the results of these experiments here. The interested reader is referred to [88] for further examples.

Discussion

For this example, the form (5.19) is a poor approximation to make. It assumes that the background errors are spatially uncorrelated and means that observations will have no effect on the analysis at surrounding grid points. When both \mathbf{B}_{zz} and \mathbf{R} are diagonal the expression for the gain matrix \mathbf{K} simplifies and the analysis equation (2.9) reduces to

$$\mathbf{z}_k^a = \mathbf{z}_k^b + \frac{\sigma_b^2}{\sigma_b^2 + \sigma_o^2}(\mathbf{y}_k - \mathbf{H}_k \mathbf{z}_k^b). \quad (5.26)$$

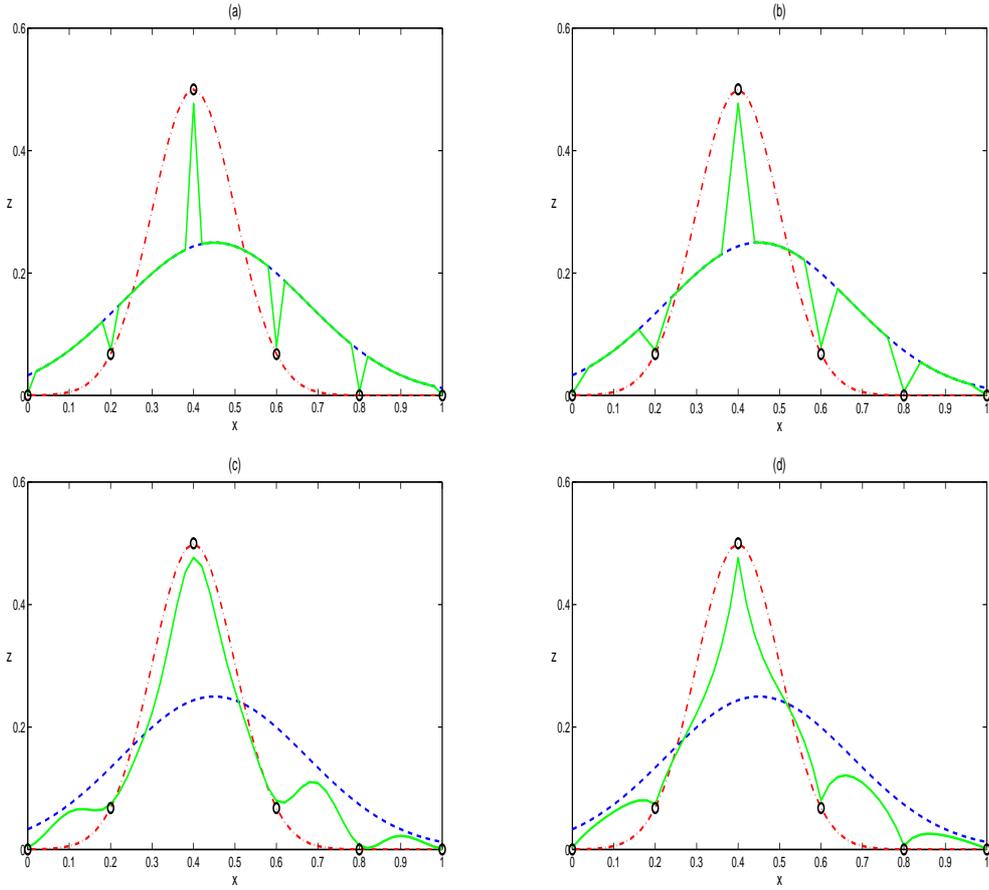


Figure 5.4: Comparison of background error covariance matrices \mathbf{B}_{zz} : **(a)** Diagonal matrix (5.19), **(b)** Tri-diagonal matrix (5.21), **(c)** Gaussian matrix (5.22), **(d)** Markov matrix (5.23). The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b by the blue dashed line and the analysis \mathbf{z}^a by the solid green line.

The gain matrix \mathbf{K} now acts like a scalar; this means that each observation only effects the value of the analysis at the grid point at which it is taken. At points where there are observations the analysis is given by a weighted average of the value of the background and observation at that point. Since we have taken $\sigma_o^2 \ll \sigma_b^2$, a relatively small weight is given to the background state and the analysis lies close to the observed value. At grid points where no observation is taken the analysis is set equal to the background, resulting in a jagged analysis curve.

With the tri-diagonal matrix (5.21) observational information is now spread to the two neighbouring grid points and so also effects the analysis at points directly adjacent to the observation location. However, unless observations are taken at at least every other grid point, this still leaves regions where the available observations have no effect on the analysis and therefore generates oscillations as above.

The full covariance matrices (5.22) and (5.23) give much better results. The presence of non-zero cross covariances mean that the matrix \mathbf{B}_{zz} now has a much larger radius of influence and the information contained in the observations is spread further into the domain. Each observation now affects more of the surrounding grid points enabling the shape of the true solution to be more accurately captured, even with relatively few observations.

However, the quality of the analysis is sensitive to changes in the value of the correlation length scale L . It is important to ensure that the background error correlations are defined on sensible scales. By varying the size of L in (5.22) and (5.23) we can vary the size of the correlations ρ_{ij} and in turn the cross covariance elements of the matrix \mathbf{B}_{zz} . When there is a mismatch between the spatiotemporal resolution of the model and the density of the observations the correlation length scale L allows us to determine how the observed information is used by controlling the amount of information smoothing.

In figures 5.4 (c) and (d), L is set equal to 0.05. Figures 5.5 and 5.6 show how the analysis changes as L is varied from 0.025 to 0.2, for matrices (5.22) and (5.23) respectively. We use the same observation combination as in the previous experiment. For the Gaussian matrix (5.22), the shape of the true solution is captured most accurately with $L = 0.1$. If L is increased to 0.2 the analysis becomes over-smoothed; the sediment hump is too wide and its peak is too low. Conversely, if L is set too small the analysis is ‘lumpy’. The Markov matrix (5.23) is less sensitive to the value of L . As with the Gaussian matrix, we see an improvement in the analysis when L is increased to 0.1 but the effect of a further increase to $L = 0.2$ is less marked.

Choosing L is a balancing act; we want to extract the maximum amount of information from the observations but we need to limit the amount of smoothing so that the analysis produced is physically accurate. When L is large the covariance (correlation) between the background errors at any pair of grid points is greater. Small scale structures are suppressed and this generates an analysis that is comparatively smooth. When L is small the background error correlations are reduced. There is less filtering of small scale structures, but the filtering of the large scale structures increases so that away from the observation points the analysis reverts to the background field. In this case each observation has a much narrower region of effect, and therefore less influence on

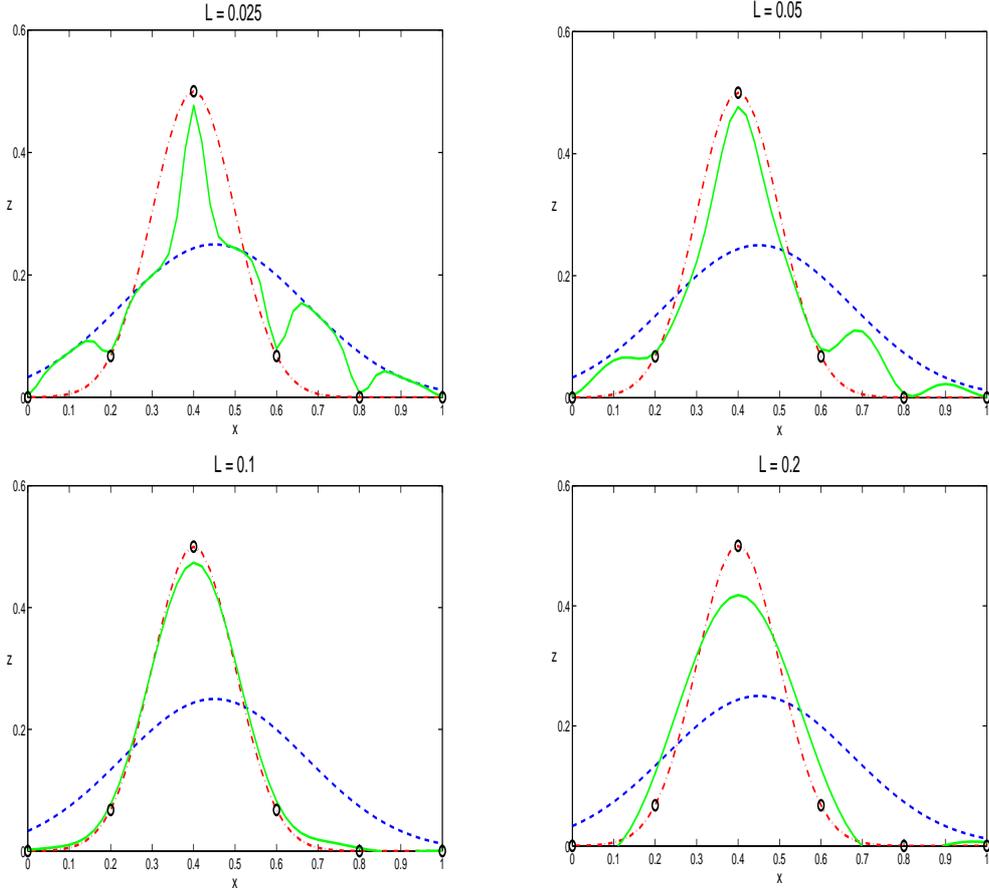


Figure 5.5: Effect of varying the correlation length scale L in the Gaussian state background error covariance matrix \mathbf{B}_{zz} (5.22). The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b by the blue dashed line and the analysis \mathbf{z}^a by the solid green line.

the analysis. In order to be able to accurately reconstruct the model state we must ensure that sufficient weight is given to the background state. The results of further experiments demonstrating how the choice of L can effect the analysis for different observation frequencies and densities are given in [88]. For a more in depth discussion of the filtering properties of the background error correlations the interested reader is referred to section 4.5 of [14].

Overall, the Gaussian matrix (5.22) with an appropriate choice of L was most effective, producing an analysis whose shape is much smoother and closer to the true solution than the other matrices. Although the analyses it produced were less smooth than the Gaussian matrix, the form (5.23) also worked well at capturing the general shape of the solution. This matrix has the additional benefit that its inverse can be calculated explicitly and has a particularly simple tri-diagonal form [89]. This is advantageous

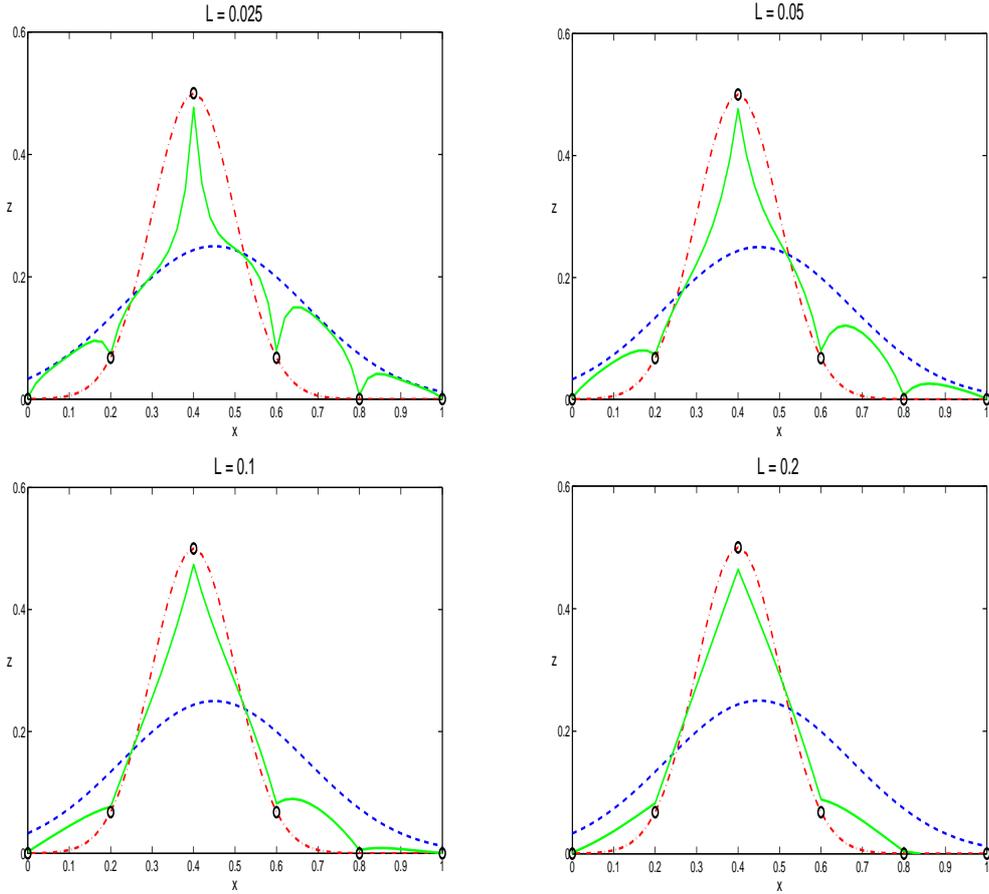


Figure 5.6: Effect of varying the correlation length scale L in the Markov state background error covariance matrix \mathbf{B}_{zz} (5.23). The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b by the blue dashed line and the analysis \mathbf{z}^a by the solid green line.

when the dimension of the system is large. For this reason, we use this matrix in the experiments presented in sections 5.5 and 5.6, section 7.1 of chapter 7 and chapter 8.

Having verified our minimisation algorithm and determined a suitable form for the state background error covariance matrix \mathbf{B}_{zz} we now move on to consider the joint state-parameter estimation problem.

5.4 Joint state-parameter estimation

We saw in the previous section the impact that the choice of the background error covariance matrix can have on the assimilation results. For the joint state-parameter estimation problem we have the additional challenge of specifying the background error

covariance matrices for the parameter vector, $\mathbf{B}_{\mathbf{pp}}$, and for the cross correlations between the state and parameter errors, $\mathbf{B}_{\mathbf{zp}}$.

5.4.1 Parameter error covariance

For our simple linear advection model (5.3), we only have a single unknown parameter, the bed celerity c ; in this case the parameter vector \mathbf{p} is a scalar. We approximate the true celerity c with \tilde{c} , where $\tilde{c} = c + \varepsilon_c$ is the estimated value at a particular time t_k . Setting $\varepsilon_{\mathbf{p}_k} = \varepsilon_c$ and using definition (3.9), we have

$$\mathbf{B}_{\mathbf{pp}_k} = E[\varepsilon_c^2] = \sigma_c^2 \quad (5.27)$$

where σ_c^2 is the parameter error variance.

We assume that parameter errors are statistically stationary and set

$$\mathbf{B}_{\mathbf{pp}_k} = \mathbf{B}_{\mathbf{pp}} = \sigma_c^2 \quad \text{for all } k, \quad (5.28)$$

i.e. the parameter error covariance matrix remains fixed for all t_k . The parameter error variance σ_c^2 is prescribed a value that represents our uncertainty in the estimated parameter.

5.4.2 Modelling the state-parameter cross covariances

To define the matrix $\mathbf{B}_{\mathbf{zp}_k}$, we need to consider the relationship between the errors in the parameter estimates $\varepsilon_{\mathbf{p}_k}^b$ and the errors in the state background $\varepsilon_{\mathbf{z}_k}^b$. As they depend on the same data, we expect them to be correlated.

One possible method for calculating these covariances is by averaging the statistics over the assimilation window, using our knowledge of the truth and background states. However, since in reality the true solution is not known, this is difficult to do in practice. For simplicity we would like this matrix to be of a functional form similar to that used for the state background error covariance matrix (5.23). As we explained in chapter 3, section 3.4, successful parameter estimation relies upon these correlations being suitably specified, so it is important to ensure that the choice of function is appropriate to the particular model application.

5.4.3 State background errors

To determine a suitable form for $\mathbf{B}_{\mathbf{z}^k}$, we first seek an approximation to the state background error $\varepsilon_{\mathbf{z}^k}^b$. We begin by considering a single realisation at a particular point x and time t . The state background error at this point, which we denote $\varepsilon_z(x, t)$, will be a combination of error in the initial condition and error in the parameter estimate. There are four possibilities: (1) known initial bathymetry and known advection velocity, (2) unknown initial bathymetry and known advection velocity, (3) known initial bathymetry and unknown advection velocity and (4) unknown initial bathymetry and unknown advection velocity. Here, we consider case 4 but note that solutions for the other three cases can be derived in a similar manner as described in [89].

We define

$$\tilde{z}(x, t) = z(x, t) + \varepsilon_z(x, t), \quad (5.29)$$

and

$$\tilde{f}(x) = f(x) + \varepsilon_z(x, 0), \quad (5.30)$$

where $\tilde{z}(x, t)$ is our approximation to the true bathymetry $z(x, t)$ at time t and $\tilde{f}(x)$ is our estimate of the true initial state $f(x) = z(x, 0)$.

From (5.5), we have the solution

$$\tilde{z}(x, t) = \tilde{f}(x - \tilde{c}t), \quad t \geq 0. \quad (5.31)$$

Using (5.29)

$$\begin{aligned} \varepsilon_z(x, t) &= \tilde{z}(x, t) - z(x, t) \\ &= \tilde{f}(x - \tilde{c}t) - f(x - ct) \\ &= \tilde{f}(x - ct - \varepsilon_c t) - f(x - ct). \end{aligned} \quad (5.32)$$

Assuming that $\varepsilon_c t$ is small and that $f(x)$ is a continuous, differentiable function, we can expand (5.32) in a Taylor series about $\tilde{f}(x - ct)$, yielding

$$\begin{aligned} \varepsilon_z(x, t) &= \tilde{f}(x - ct - \varepsilon_c t) - f(x - ct) \\ &= \left[\tilde{f}(x - ct) - \varepsilon_c t \tilde{f}'(x - ct) + \frac{\varepsilon_c^2}{2!} t^2 \tilde{f}''(x - ct) - \dots \right] - f(x - ct) \\ &= \varepsilon_z(x - ct, 0) - \varepsilon_c t \tilde{f}'(x - ct) + O\left((\varepsilon_c t)^2\right). \end{aligned} \quad (5.33)$$

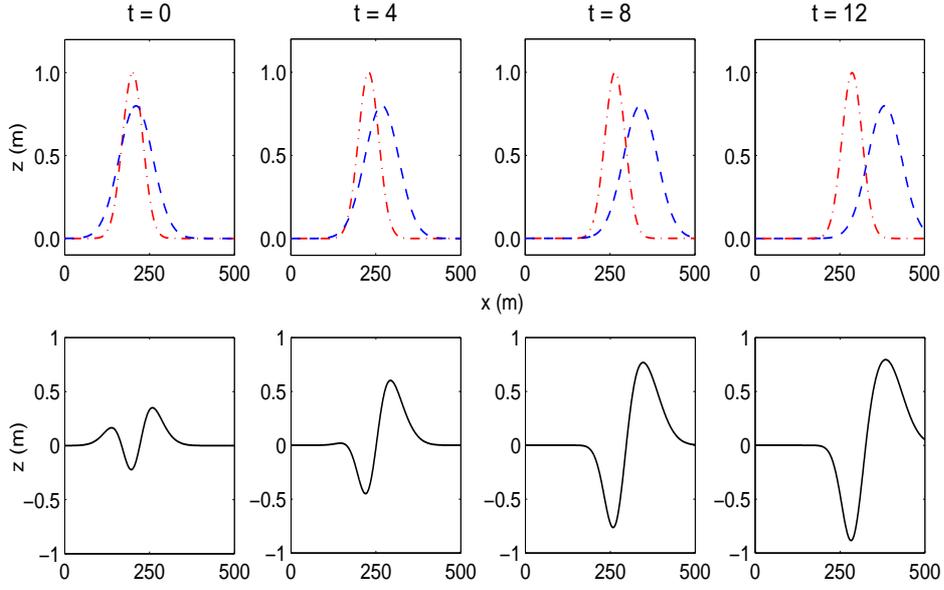


Figure 5.7: Linear advection model: unknown initial bathymetry and unknown advection velocity. **Top:** true bathymetry $z(x, t)$ (dot-dash red line) and predicted model bathymetry $\tilde{z}(x, t)$ (dashed blue line). **Bottom:** background error $\varepsilon_z(x, t) = \tilde{z}(x, t) - z(x, t)$.

If we further assume that the errors $\varepsilon_z(x, 0)$ are smooth, we can use (5.30) to rewrite (5.33) as

$$\varepsilon_z(x, t) = \varepsilon_z(x - ct, 0) - \varepsilon_c t f'(x - ct) + \dots \quad (5.34)$$

The solution (5.31) and its error (5.34) are illustrated in figure 5.7. As time increases the higher order terms in (5.34) dominate. Incorrect specification of the advection velocity causes a phase error that grows with time, overshadowing the error in the initial bed profile. We note that the error is similar in character to the derivative of the initial state $f'(x)$ but becomes amplified as time increases.

5.4.4 State-parameter cross covariance

Since the parameter error $\varepsilon_{\mathbf{p}}^b = \varepsilon_c$ is scalar, the cross covariance matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}_k}$ will be a vector of length m . Using the definition (3.9), we have

$$\mathbf{B}_{\mathbf{z}\mathbf{p}_k} = E \left[\varepsilon_{\mathbf{z}_k}^b \left(\varepsilon_{\mathbf{p}_k}^b \right)^T \right] = E \left[\varepsilon_c \varepsilon_{\mathbf{z}_k}^b \right] = \begin{bmatrix} E(\varepsilon_c \varepsilon_{z_{k,1}}) \\ E(\varepsilon_c \varepsilon_{z_{k,2}}) \\ \vdots \\ E(\varepsilon_c \varepsilon_{z_{k,m}}) \end{bmatrix}. \quad (5.35)$$

Here, $\varepsilon_{z_k,j}$ is the j th component of the error vector $\boldsymbol{\varepsilon}_{\mathbf{z}_k}^b$, representing the background error associated with the state background estimate \mathbf{z}_k^b at the grid point x_j at time t_k . Element $b_{zp_k,j} = E(\varepsilon_c \varepsilon_{z_k,j})$ defines the covariance between ε_c and $\varepsilon_{z_k,j}$.

Ideally, we would like to apply the same principle as was done for the state and parameter background error covariance matrices $\mathbf{B}_{\mathbf{z}\mathbf{z}}$ and $\mathbf{B}_{\mathbf{p}\mathbf{p}}$, and assume that the cross covariances are stationary and define $\mathbf{B}_{\mathbf{z}\mathbf{p}_k} = \mathbf{B}_{\mathbf{z}\mathbf{p}}$ using a function that has a fixed structure for all t_k . The error analysis in section (5.4.3) showed that for sufficiently small t , the background error at the point x at time t will be linearly related to the value of the derivative of the initial state at the starting point $x_0 = x - ct$. Guided by this result we propose the following time independent approximation for the background error

$$\varepsilon_{z_j} = -\varepsilon_c f'(x_j), \quad j = 1, \dots, m. \quad (5.36)$$

To obtain an expression for the cross covariances (5.35), we multiply (5.36) by ε_c and take the expected value over many realisations, to give

$$\begin{aligned} b_{zp_j} &= -E(\varepsilon_c^2 f'(x_j)) \\ &= -E(\varepsilon_c^2) f'(x_j) \\ &= -\sigma_c^2 f'(x_j). \end{aligned} \quad (5.37)$$

Since by the nature of the problem the true bathymetry is unknown, we cannot compute the derivative $f'(x)$. In order to evaluate the expression (5.37), we approximate using the derivative of the background bathymetry $\tilde{f}(x)$ at initial time t_0 . The matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}}$ entries then become

$$b_{zp_j} = -\sigma_c^2 \tilde{f}'(x_j), \quad j = 1, \dots, m. \quad (5.38)$$

5.5 Assimilation experiments

Given an approximate velocity \tilde{c} and starting from a perturbed initial state, we wish to investigate whether our augmented data assimilation scheme is able to deliver both an accurate model bathymetry and an accurate estimate of the true advection velocity c . The analytic solution (5.5) is used to evaluate the performance of the method.

The augmented state vector takes the form

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{z}_k \\ c_k \end{pmatrix}. \quad (5.39)$$

The assimilation process is carried out sequentially as described in section 2.3.1, algorithm 2.1, with a new set of observations being assimilated at regular intervals. At the end of each assimilation cycle, the analysis is integrated forward using the model to obtain the background state for the next analysis time.

5.5.1 True model state

We generate a reference solution for z by solving (5.3) for $c > 0$ with initial data given by the Gaussian function (5.6) with $\alpha = 1.0$, $\beta = 200$ and $\gamma = 30$. For these experiments we fix the true advection velocity at a physically realistic value of $c = 0.002 \text{ ms}^{-1}$. The model was sampled on a regular grid with spacing $\Delta x = 1.0 \text{ m}$ and time step $\Delta t = 15$ minutes.

5.5.2 Observations

We assume that we have perfect, direct observations of z at r evenly spaced points along the model domain. These are generated using the analytic solution (5.5) and chosen so that they coincide with the model grid. The location and temporal frequency of the observations is specified at the start of each experiment and remains fixed throughout the assimilation. The augmented observation operator $\tilde{\mathbf{h}}$ is therefore linear and takes the form of a constant $r \times (m + 1)$ matrix of zeros and ones. Hence, the observation vector at time t_k is given by

$$\mathbf{y}_k = \tilde{\mathbf{H}}\mathbf{w}_k^t = \mathbf{H}\mathbf{z}_k^t. \quad (5.40)$$

The observation error covariance matrix is given by (5.18) with $\sigma_o^2 = 0.1$. The use of perfect observations is common practice in the preliminary testing of data assimilation schemes with synthetic data. Specifying a non-zero observation error variance allows the impact of the accuracy of the observations to be investigated without actually adding noise [46], [64].

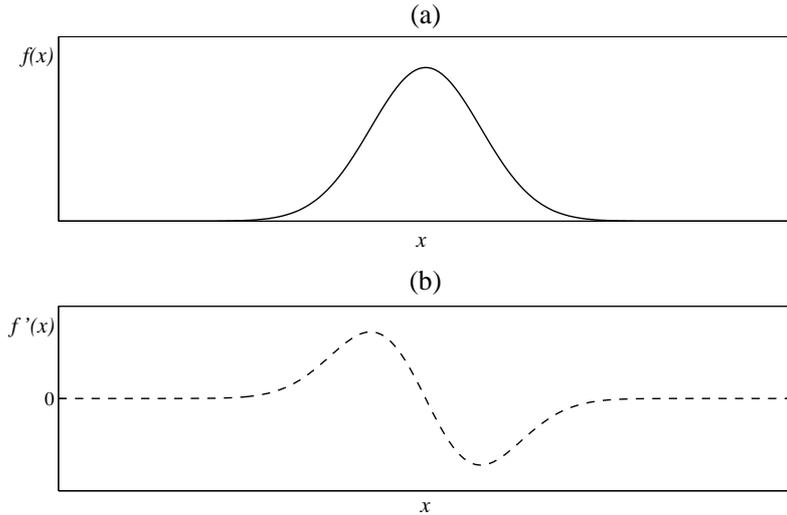


Figure 5.8: Schematic showing (a) a Gaussian function $f(x) = \alpha e^{-(x-\beta)^2/2\gamma^2}$ and (b) its derivative $f'(x)$.

5.5.3 Background error covariance

The initial model bathymetry is also taken to be a Gaussian but is rescaled so that it is slightly shorter and wider and offset from the true initial state. In this example, we overestimate the true advection velocity, setting $\tilde{c} = 0.004 \text{ ms}^{-1}$. The state and parameter background error covariance matrices \mathbf{B}_{zz} and \mathbf{B}_{pp} are defined by (5.23) and (5.28) with $\sigma_b^2 = 1.0$ and $\sigma_c^2 = 0.001$ respectively. The matrix \mathbf{B}_{zp} is calculated at the start of the assimilation using (5.38) and remains constant throughout. Since we are assuming that the initial background bathymetry has a Gaussian structure, its derivative will be of the form shown in figure 5.8.

5.5.4 Results

Figure 5.9 shows the state analysis produced with initial parameter estimate $\tilde{c} = 0.004 \text{ ms}^{-1}$. Observations were taken at $25\Delta x$ intervals and assimilated every hour. We found that although qualitatively the state analysis is close to the truth (c.f. figure 5.7 in which the model is run without data assimilation) the scheme was unable to recover the value of the true advection velocity $c = 0.002 \text{ ms}^{-1}$. The corresponding parameter updates are shown in figure 5.10 along with those for a second test case in which the velocity is initially under estimated ($\tilde{c} = 0.001 \text{ ms}^{-1}$). The estimates do eventually appear to

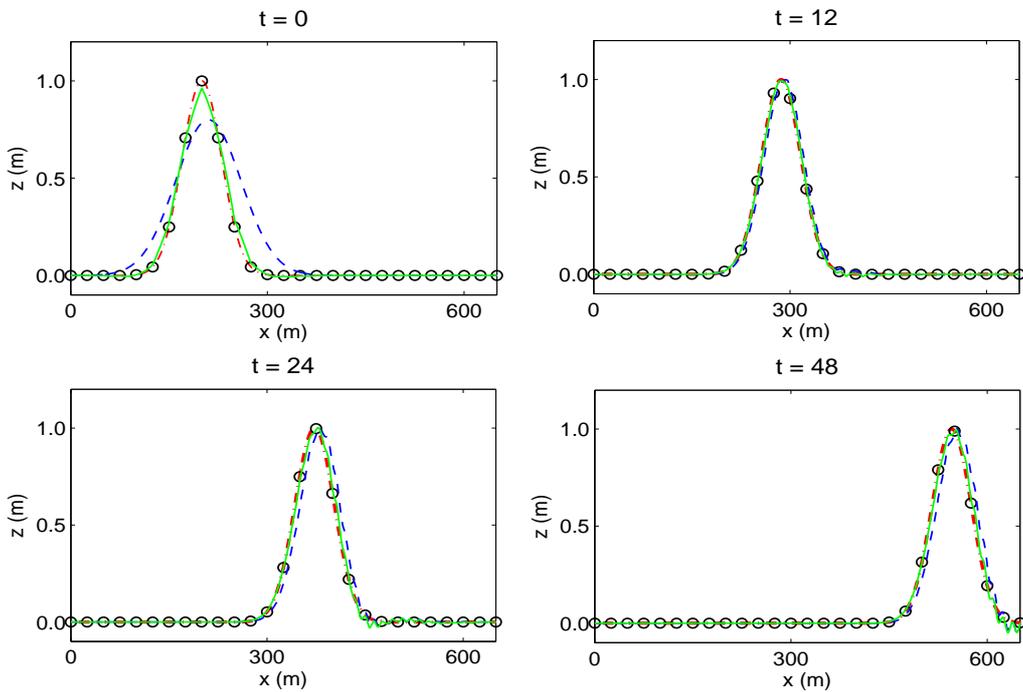


Figure 5.9: Assimilation with static state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.38): state analysis at times $t = 0, 12, 24$ & 48 hours, for initial parameter estimate $\tilde{c} = 0.004 \text{ ms}^{-1}$. The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

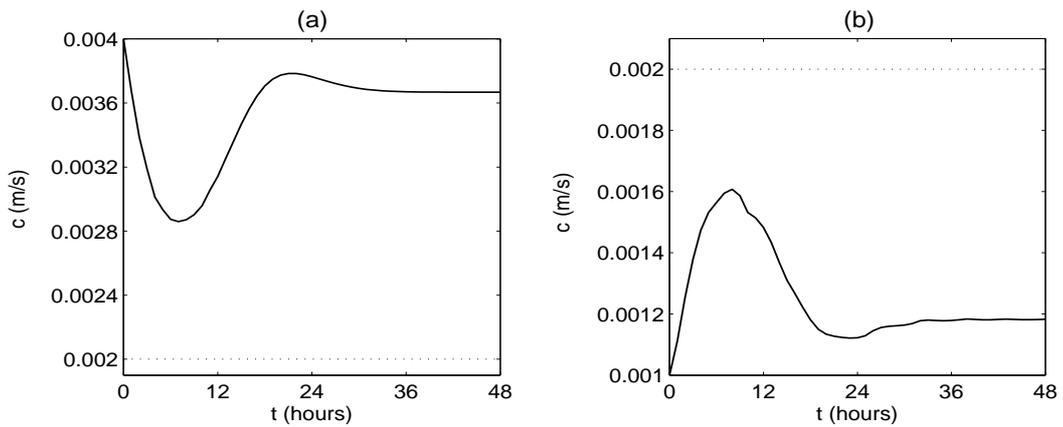


Figure 5.10: Assimilation with static state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.38): updating of parameter c for initial estimates (a) $\tilde{c} = 0.004 \text{ ms}^{-1}$ (b) $\tilde{c} = 0.001 \text{ ms}^{-1}$. The dotted black line indicates the true parameter value $c = 0.002 \text{ ms}^{-1}$.

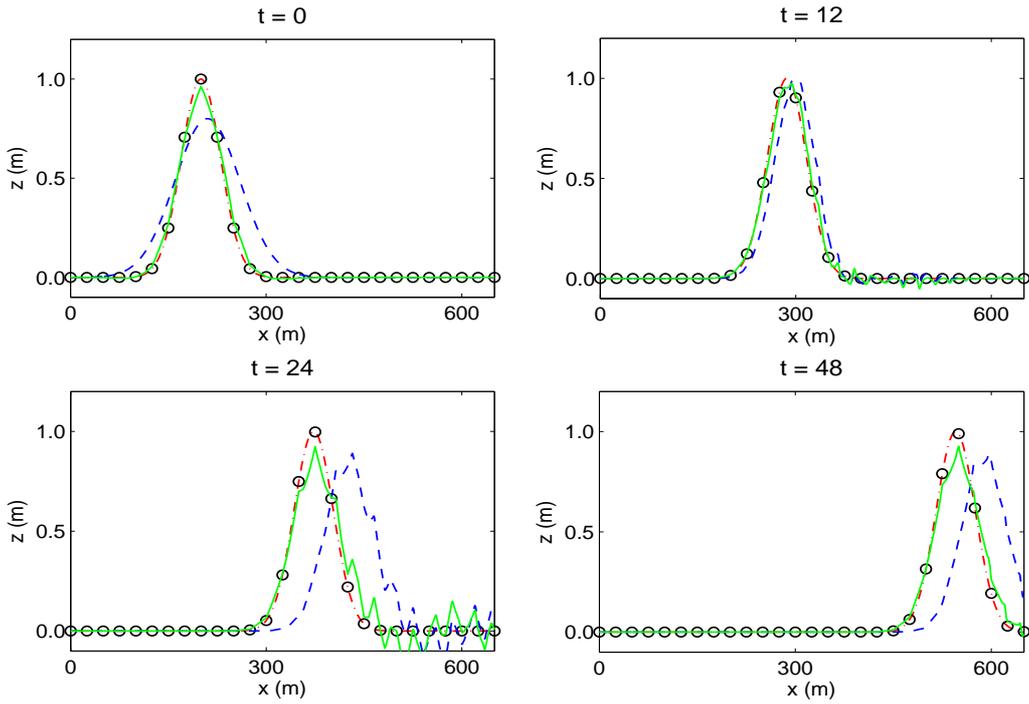


Figure 5.11: Assimilation with static state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.38): state analysis at times $t = 0, 12, 24$ & 48 hours, for initial parameter estimate $\tilde{c} = 0.02 \text{ ms}^{-1}$. The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

converge but towards incorrect values. Because the error in the estimated advection velocity is relatively small and the time between successive observations is relatively short, the state estimation alone is sufficient enough to keep the model bathymetry on track. When the error in the estimate of the advection velocity c is large, the model diverges from the true state and the quality of the analysis is poor as in illustrated in figure 5.11 for the case $\tilde{c} = 0.02 \text{ ms}^{-1}$.

We know that the measurements of the bed height z provide information for refining the estimate of c via the covariance matrix $\mathbf{B}_{z\mathbf{p}}$. The results of these experiments indicate that the fixed approximation (5.38) does not provide an adequate enough representation of these cross covariances to enable reliable parameter updating.

5.6 An alternative approach

The expression for the state background error (5.34) derived above is based on a scenario where there is no data assimilation, and therefore assumes that the form of the model

approximation of the state, $\tilde{z}(x, t) = \tilde{f}(x - \tilde{c}t)$, and estimated advection velocity, \tilde{c} , remain the same for all time. With data assimilation, the bed profile and speed change as the background $\tilde{z}(x, t)$ and parameter estimate \tilde{c} are updated by the assimilation of observations at each new analysis time. Thus we will have a different \tilde{f} and \tilde{c} at the start of each new model integration. As $\tilde{f}(x)$ and \tilde{c} change so too will the background errors ε_{z_k} and ε_c and the correlation between them.

To take account of the fact that the background-parameter error cross covariances will change with each new analysis we consider using a time varying approximation for the matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}}$. Although the exact structure of the errors will vary as the data assimilation updates $\tilde{f}(x)$ and \tilde{c} , our practical experiments show that the background errors can still broadly be described by the derivative function $f'(x - ct)$. Re-introducing time dependence into the approximation (5.36), we have

$$\varepsilon_{z_k, j} = -\varepsilon_c f'(x_j - ct_k), \quad j = 1, \dots, m. \quad (5.41)$$

Since $f'(x)$ and c are unknown we cannot evaluate the expression (5.41). Instead, we substitute f' with the derivative of the background \tilde{f}' and replace $(x - ct)$ with $(x - \hat{s}_k)$, where \hat{s}_k is a time dependent value that represents the distance travelled from the starting point x_0 at time t_k . This yields the approximation

$$\varepsilon_{z_k, j} = -\varepsilon_c \tilde{f}'(x_j - \hat{s}_k), \quad j = 1, \dots, m. \quad (5.42)$$

The shape of the function \tilde{f}' will change each time the background state \tilde{f} is updated by the assimilation. For ease of computation, rather than recalculate \tilde{f}' each time the matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}}$ needs updating (i.e. at each new assimilation time) we use a fixed structure given by the derivative of the initial background bathymetry $\tilde{f}'(x)$ at t_0 , as was done for the static covariances. However, rather than keeping the spatial location of the derivative fixed, it is now translated across the domain as the model steps forward. Its position at time t_k is determined by the value \hat{s}_k . To ensure that the error covariances remain in phase with the model state, \hat{s}_k is chosen such that $\tilde{f}'(x - \hat{s}_k)$ is centred at the maximum of the current model predicted bathymetry $\tilde{z}(x, t)$.

Proceeding as above, we multiply (5.42) by ε_c and take the expected value over many realisations. This gives the time dependent matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}_k}$ elements

$$b_{z\mathbf{p}_k, j} = -\sigma_c^2 \tilde{f}'(x_j - \hat{s}_k), \quad j = 1, \dots, m. \quad (5.43)$$

5.6.1 Results

We repeat the experiments of section 5.5, using our new approximation for \mathbf{B}_{zp} . The submatrices \mathbf{B}_{zz} and \mathbf{B}_{pp} remain fixed but the cross covariances are now updated at each new analysis time using the approximation (5.43).

Figure 5.12 shows the parameter updates for initial parameter estimates (a) $\tilde{c} = 0.004 \text{ ms}^{-1}$ and (b) $\tilde{c} = 0.001 \text{ ms}^{-1}$. The state analysis for case (a) is shown in figure 5.13, the state analysis for (b) is not shown as it is qualitatively similar. In both cases, the scheme converges after around 24 hours, managing to successfully recover the true value of c to a high level of accuracy. We found that the rate of convergence of the estimates could be increased by increasing the parameter error variance. Figure 5.14 shows the effect of doubling the error variance from $\sigma_c^2 = 0.001$ to $\sigma_c^2 = 0.002$.

Figure 5.15 (a) and (b) show two further test cases in which we assume a much greater error in the initial value of c . Here, we set (a) $\tilde{c} = 0.02 \text{ ms}^{-1}$ and (b) $\tilde{c} = 0.0002 \text{ ms}^{-1}$. The parameter error variance σ_c^2 is increased to $\sigma_c^2 = 0.005$ to reflect the larger uncertainty in c . Again, the scheme manages to successfully recover the true value of c . As a result, the model becomes a much better approximation and, thus, produces more accurate estimates of the true bathymetry. The impact on the state analysis is particularly noticeable for the case $\tilde{c} = 0.02 \text{ ms}^{-1}$, and can be seen by comparing figure 5.16 with figure 5.11 showing the analysis produced with the static matrix \mathbf{B}_{zp} for the same example.

The experiments were repeated for a range of both over and under estimated c values, investigating the sensitivity of the parameter estimates to varying background guesses, observation combinations and observation noise. The quality of the analysis is highly dependent on the accuracy of the information fed into the assimilation algorithm. The speed of convergence and accuracy of the parameter estimates varies depending on the quality of the background state, the location and spatial frequency of the observations, the level of observational noise and the time between successive assimilations. We do not present the results of these experiments here but refer the reader to [89] and [90] for further details and discussion.

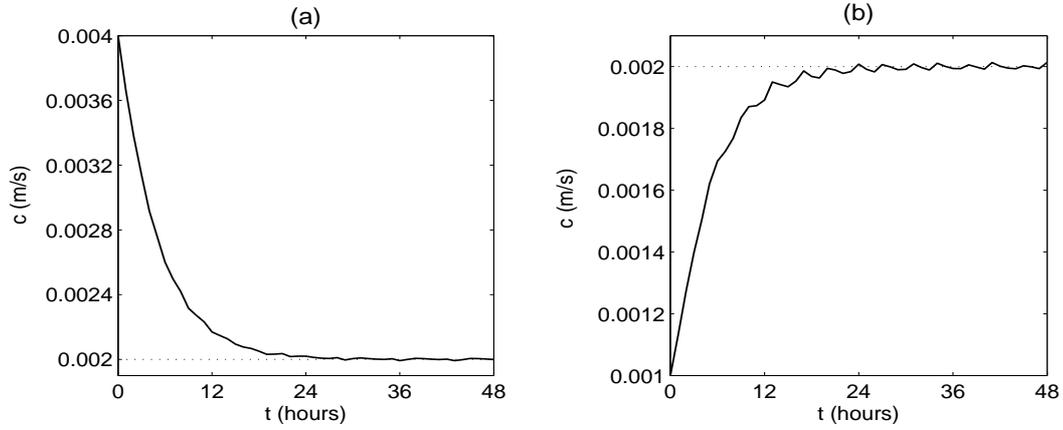


Figure 5.12: Assimilation with time varying state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.43): updating of parameter c for initial estimates (a) $\tilde{c} = 0.004 \text{ ms}^{-1}$ (b) $\tilde{c} = 0.001 \text{ ms}^{-1}$. The dotted black line indicates the true parameter value $c = 0.002 \text{ ms}^{-1}$.

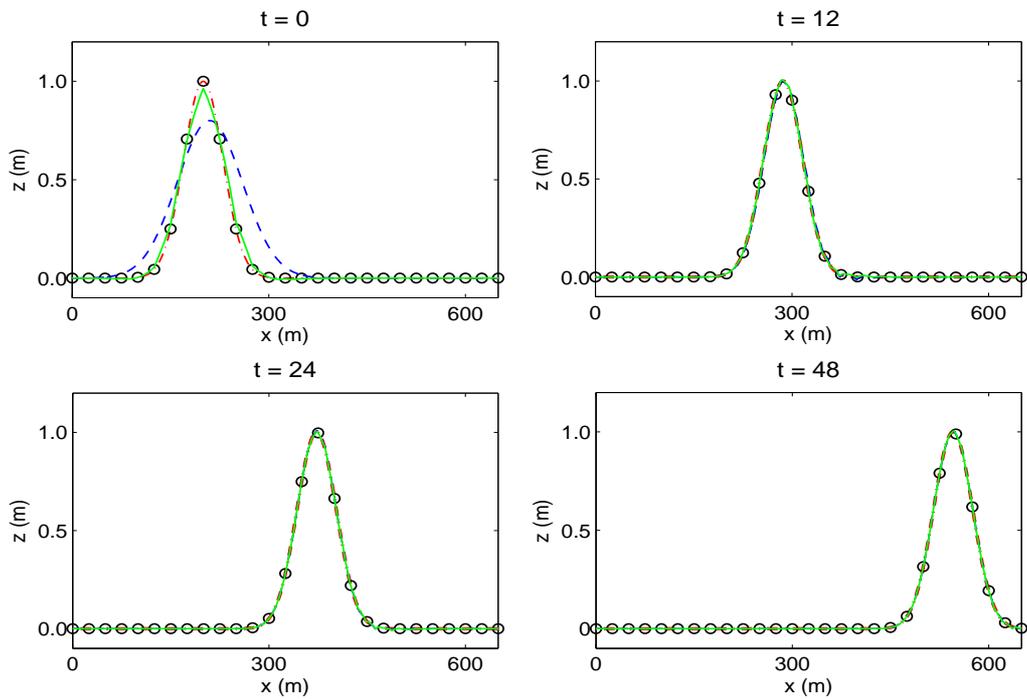


Figure 5.13: Assimilation with time varying state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.43): state analysis at times $t = 0, 12, 24$ & 48 hours, for initial parameter estimate $\tilde{c} = 0.004 \text{ ms}^{-1}$. The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

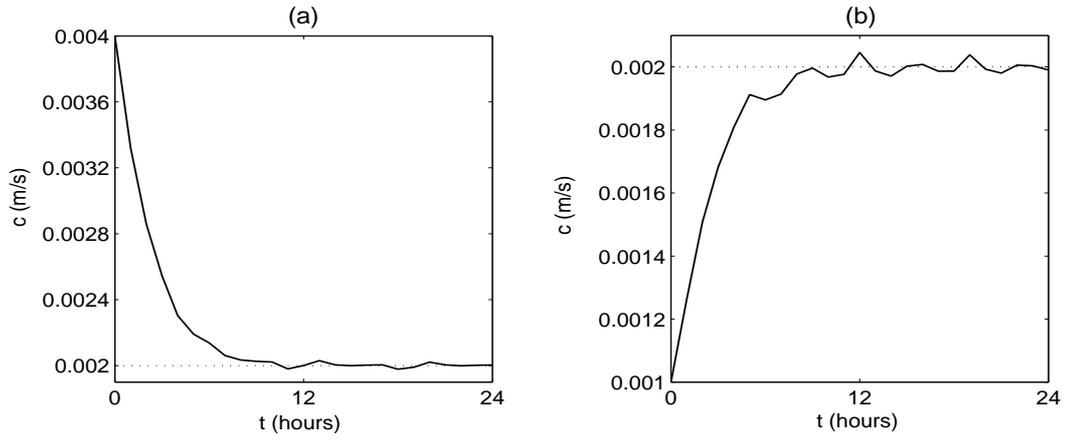


Figure 5.14: Assimilation with time varying state-parameter cross covariance matrix \mathbf{B}_{zp} (5.43): updating of parameter c for initial estimates **(a)** $\tilde{c} = 0.004 \text{ ms}^{-1}$ **(b)** $\tilde{c} = 0.001 \text{ ms}^{-1}$, $\sigma_c^2 = 0.002$. The dotted black line indicates the true parameter value $c = 0.002 \text{ ms}^{-1}$.

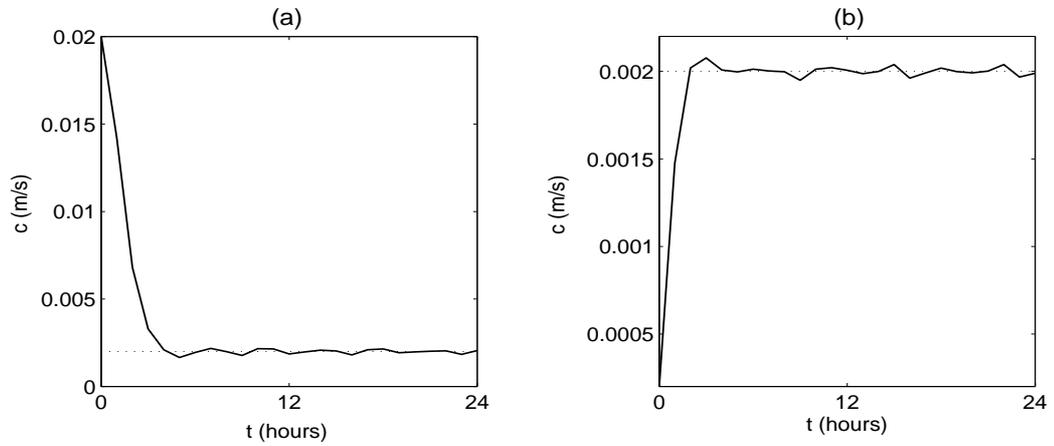


Figure 5.15: Assimilation with time varying state-parameter cross covariance matrix \mathbf{B}_{zp} (5.43): updating of parameter c for initial estimates **(a)** $\tilde{c} = 0.02 \text{ ms}^{-1}$ **(b)** $\tilde{c} = 0.0002 \text{ ms}^{-1}$, $\sigma_c^2 = 0.005$. The dotted black line indicates the true parameter value $c = 0.002 \text{ ms}^{-1}$.

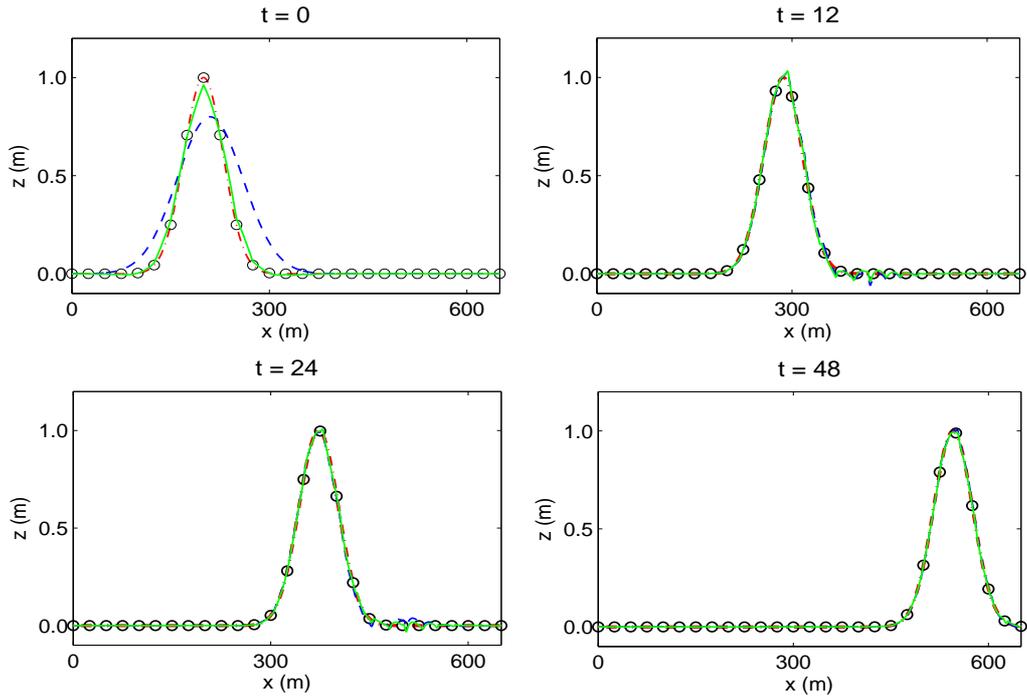


Figure 5.16: Assimilation with time varying state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}}$ (5.43): state analysis at times $t = 0, 12, 24$ & 48 hours, for initial parameter estimate $\tilde{c} = 0.02\text{ms}^{-1}$. The red dot-dash line represents the true bathymetry \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

5.7 Summary

In this chapter we used a simple 1D linear advection model to investigate some of the issues associated the design of a new assimilation scheme. We gave particular consideration to the role of the background error covariances and used a series of simple experiments to illustrate the effect that the choice of background error covariance matrix can have on the analysis in state estimation. We then examined ways of specifying the state-parameter cross covariances needed for joint state-parameter estimation. It was found that in order to recover the true parameter values it was necessary to use a flow dependent approximation. However, it does not appear to be necessary to explicitly propagate the full system covariance matrix. We were able to get accurate estimates of both the parameters and state variables by combining a time varying approximation of the state-parameter cross covariance matrix $\mathbf{B}_{z\mathbf{p}_k}$ with an empirical, static representation of the state background error covariance \mathbf{B}_{zz} . In the next chapter, we consider this result further. We seek to develop a method for approximating the augmented matrix \mathbf{B}_k that can be applied to a range of general dynamical system models.

Chapter 6

A hybrid approach

In chapter 5 we highlighted the role of the background error covariance matrix \mathbf{B}_k . For concurrent state-parameter estimation, it is particularly important that the state-parameter cross covariances are given a good a priori specification. The results of our experiments with a simple 1D linear advection model showed that in order to be able to reliably update the model parameters the cross covariance matrix \mathbf{B}_{zp} needed to have a time varying structure. Here, we were able to use our a priori knowledge of the parameter and of the behaviour of the solution to derive a suitable form for this matrix, but in practice this type of information is likely to be unavailable or at best limited. Ideally, the full augmented background error covariance matrix \mathbf{B}_k should be flow-dependent and propagated forward from the previous assimilation time. However such an approach is expensive and impractical for large environmental systems. In general, a much simpler method of matrix updating will be required. A key part of our findings from chapter 5 was that, provided the state-parameter cross covariances were well defined, it was not necessary to propagate the state background error covariance matrix \mathbf{B}_{zz} . We now wish to formalise these results and establish a generalised method for approximating the augmented background error covariance matrix \mathbf{B}_k that is applicable across a range of dynamical system models.

In section 6.1 we consider how to prescribe the state-parameter cross covariance matrix \mathbf{B}_{zp_k} for a general model. Using the results of the previous chapter we propose a hybrid approach to approximating the matrix \mathbf{B}_k ; combining a flow dependent approximation

of the state-parameter cross covariance matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}_k}$ with a static representation of the state background error covariance $\mathbf{B}_{\mathbf{z}\mathbf{z}}$ and parameter error covariance $\mathbf{B}_{\mathbf{p}\mathbf{p}}$. In section 6.2 we discuss aspects relating to practical implementation of this new method and in section 6.3 we outline some of its advantages in comparison to existing techniques. We finish by describing how this new approach can be incorporated into a sequential 3D-Var data assimilation algorithm in section 6.4.

6.1 Formulation

From chapter 3, section 3.4.2, we have the discrete nonlinear time invariant augmented system forecast model

$$\mathbf{w}_{k+1}^f = \tilde{\mathbf{f}}(\mathbf{w}_k^a) \quad k = 0, 1, \dots \quad (6.1)$$

where

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{z}_k \\ \mathbf{p}_k \end{pmatrix} \in \mathbb{R}^{m+q} \quad \text{and} \quad \tilde{\mathbf{f}}(\mathbf{w}_k) = \begin{pmatrix} \mathbf{f}(\mathbf{z}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix}, \quad (6.2)$$

with $\tilde{\mathbf{f}} : \mathbb{R}^{m+q} \rightarrow \mathbb{R}^{m+q}$ and $\mathbf{f} : \mathbb{R}^m \rightarrow \mathbb{R}^m$.

One option for evolving the background error covariances would be to use the EKF error covariance forecast equation (2.21). We can rewrite (2.21) in terms of our augmented system as

$$\mathbf{P}_{k+1}^f = \mathbf{F}_k \mathbf{P}_k^a \mathbf{F}_k^T, \quad (6.3)$$

where now $\mathbf{P}_{k+1}^f \in \mathbb{R}^{(m+q) \times (m+q)}$ and

$$\begin{aligned} \mathbf{F}_k &= \left. \frac{\partial \tilde{\mathbf{f}}}{\partial \mathbf{w}} \right|_{\mathbf{w}_k^a} = \left(\begin{array}{cc} \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{z}} & \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{p}} \\ \frac{\partial \mathbf{p}}{\partial \mathbf{z}} & \frac{\partial \mathbf{p}}{\partial \mathbf{p}} \end{array} \right) \bigg|_{\mathbf{z}_k^a, \mathbf{p}_k^a} \\ &= \left(\begin{array}{cc} \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{z}} & \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{p}} \\ \mathbf{0} & \mathbf{I} \end{array} \right) \bigg|_{\mathbf{z}_k^a, \mathbf{p}_k^a} \end{aligned} \quad (6.4)$$

is the Jacobian of the augmented system forecast model evaluated at the current analysis state \mathbf{w}_k^a (see appendix A).

The augmented forecast error covariance matrix (6.3) can be partitioned as follows

$$\mathbf{P}_{k+1}^f = \begin{pmatrix} \mathbf{P}_{\mathbf{z}\mathbf{z}_{k+1}}^f & \mathbf{P}_{\mathbf{z}\mathbf{p}_{k+1}}^f \\ (\mathbf{P}_{\mathbf{z}\mathbf{p}_{k+1}}^f)^T & \mathbf{P}_{\mathbf{p}\mathbf{p}_{k+1}}^f \end{pmatrix} \quad (6.5)$$

where $\mathbf{P}_{\mathbf{z}\mathbf{z}_{k+1}}^f \in \mathbb{R}^{m \times m}$ is the forecast error covariance matrix for the state vector \mathbf{z}_{k+1}^f at time t_{k+1} , $\mathbf{P}_{\mathbf{p}\mathbf{p}_{k+1}}^f \in \mathbb{R}^{q \times q}$ is the covariance matrix describing the errors in the parameter vector \mathbf{p}_{k+1}^f and $\mathbf{P}_{\mathbf{z}\mathbf{p}_{k+1}}^f \in \mathbb{R}^{m \times q}$ is the covariance matrix for the cross correlations between the the forecast errors in the state and parameter vectors.

Using the EKF forecast equation (6.3), we consider the form of the forecast error covariance matrix for a single step of the filter. Suppose we start at time t_k with the augmented analysis error covariance matrix

$$\mathbf{P}_k^a = \begin{pmatrix} \mathbf{P}_{\mathbf{z}\mathbf{z}_k}^a & \mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a \\ (\mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a)^T & \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a \end{pmatrix} \in \mathbb{R}^{(m+q) \times (m+q)}, \quad (6.6)$$

where $\mathbf{P}_{\mathbf{z}\mathbf{z}}^a \in \mathbb{R}^{m \times m}$, $\mathbf{P}_{\mathbf{z}\mathbf{p}}^a \in \mathbb{R}^{m \times q}$ and $\mathbf{P}_{\mathbf{p}\mathbf{p}}^a \in \mathbb{R}^{q \times q}$.

If we denote

$$\mathbf{M}_k = \left. \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{z}} \right|_{\mathbf{z}_k^a, \mathbf{p}_k^a} \quad \text{and} \quad \mathbf{N}_k = \left. \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial \mathbf{p}} \right|_{\mathbf{z}_k^a, \mathbf{p}_k^a}, \quad (6.7)$$

where $\mathbf{M}_k \in \mathbb{R}^{m \times m}$ and $\mathbf{N}_k \in \mathbb{R}^{m \times q}$, we can re-write (6.4) as

$$\mathbf{F}_k = \begin{pmatrix} \mathbf{M}_k & \mathbf{N}_k \\ \mathbf{0} & \mathbf{I} \end{pmatrix}. \quad (6.8)$$

The forecast error covariance matrix (6.3) at time t_{k+1} is then given by

$$\mathbf{P}_{k+1}^f = \begin{pmatrix} \mathbf{M}_k & \mathbf{N}_k \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{P}_{\mathbf{z}\mathbf{z}_k}^a & \mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a \\ (\mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a)^T & \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a \end{pmatrix} \begin{pmatrix} \mathbf{M}_k^T & \mathbf{0} \\ \mathbf{N}_k^T & \mathbf{I} \end{pmatrix}, \quad (6.9)$$

which yields

$$\begin{aligned}
\mathbf{P}_{\mathbf{z}\mathbf{z}_{k+1}}^f &= \mathbf{M}_k \mathbf{P}_{\mathbf{z}\mathbf{z}_k}^a \mathbf{M}_k^T + \mathbf{N}_k (\mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a)^T \mathbf{M}_k^T + \mathbf{M}_k \mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a \mathbf{N}_k^T + \mathbf{N}_k \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a \mathbf{N}_k^T, \\
\mathbf{P}_{\mathbf{z}\mathbf{p}_{k+1}}^f &= \mathbf{M}_k \mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a + \mathbf{N}_k \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a, \\
\mathbf{P}_{\mathbf{p}\mathbf{p}_{k+1}}^f &= \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a.
\end{aligned} \tag{6.10}$$

We do not want to recalculate the full matrix (6.9) at every time step. The results of chapter 5 suggest that it is not necessary to propagate the state background error covariance matrix $\mathbf{B}_{\mathbf{z}\mathbf{z}}$. We therefore substitute the EKF state forecast error covariance matrix $\mathbf{P}_{\mathbf{z}\mathbf{z}_{k+1}}^f$ with a conventional 3D-Var static approximation

$$\mathbf{P}_{\mathbf{z}\mathbf{z}_k}^f = \mathbf{B}_{\mathbf{z}\mathbf{z}} \quad \text{for all } k. \tag{6.11}$$

We make the same assumption for the parameter error covariances and set

$$\mathbf{P}_{\mathbf{p}\mathbf{p}_k}^f = \mathbf{B}_{\mathbf{p}\mathbf{p}} \quad \text{for all } k. \tag{6.12}$$

To determine a method for the evolution of the state-parameter background error covariances we focus on the off diagonal blocks of (6.9). The EKF forecast state-parameter cross covariance at time t_{k+1} is given by

$$\mathbf{P}_{\mathbf{z}\mathbf{p}_{k+1}}^f = \mathbf{M}_k \mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a + \mathbf{N}_k \mathbf{P}_{\mathbf{p}\mathbf{p}_k}^a. \tag{6.13}$$

Evaluating (6.13) at every time step would introduce unwanted complexity into our scheme. We want to simplify as far as possible; we want a method for approximating the matrix $\mathbf{B}_{\mathbf{z}\mathbf{p}_k}$ that provides a suitable description of the state-parameter cross covariances but that is also straightforward to compute. We take equation (6.13) as a guide but propose a much simpler updating.

It is not unreasonable to assume that the initial background state and parameter errors are uncorrelated. Setting $\mathbf{P}_{\mathbf{z}\mathbf{p}_k}^a = \mathbf{0}$ in (6.13), using (6.12) and returning to our 3D-Var notation, gives us the following approximation for the state-parameter cross covariance matrix

$$\mathbf{B}_{\mathbf{z}\mathbf{p}_{k+1}} = \mathbf{N}_k \mathbf{B}_{\mathbf{p}\mathbf{p}}. \tag{6.14}$$

The augmented background error covariance matrix now takes the following hybrid form

$$\mathbf{B}_{k+1} = \begin{pmatrix} \mathbf{B}_{\mathbf{z}\mathbf{z}} & \mathbf{N}_k \mathbf{B}_{\mathbf{p}\mathbf{p}} \\ \mathbf{B}_{\mathbf{p}\mathbf{p}} \mathbf{N}_k^T & \mathbf{B}_{\mathbf{p}\mathbf{p}} \end{pmatrix}. \tag{6.15}$$

In other words, all elements of the background error covariance matrix (6.15) are kept fixed except the off-diagonal blocks which are updated by recalculating the matrix \mathbf{N}_k , where \mathbf{N}_k is the Jacobian of the state forecast model with respect to the parameters, as defined in (6.7). This enables us to capture the flow dependent nature of the state-parameter cross covariances without having to explicitly evolve the full system matrix.

6.2 Practical implementation

Before we can apply our new method in practice we need to prescribe the matrices \mathbf{B}_{zz} and \mathbf{B}_{pp} . We also need to establish a method for computing \mathbf{N}_k . For a simple model, it may be possible to derive an analytic expression for the Jacobian matrix \mathbf{N}_k , or an explicit computational form may be available as is the case in chapter 7. In larger, more complex models \mathbf{N}_k could be computed using finite differences or an automatic differentiation tool [31], [78].

The appropriate choice for \mathbf{B}_{zz} will depend on the particular model application. As we discussed in chapter 5, section 5.3, the simplest approach is to define \mathbf{B}_{zz} using an analytic correlation function. Alternatively, a more sophisticated covariance representation for \mathbf{B}_{zz} can be obtained using one of the empirical techniques discussed in [2], [24].

Specification of the parameter background error covariance matrix \mathbf{B}_{pp} requires some level of a priori knowledge of the parameter error statistics. The individual parameter error variances should reflect our uncertainty in the model parameters but must also be chosen so as to ensure that the parameter updates are realistic and consistent with the scales of the model. For models with more than one uncertain parameter, we will also need to consider the correlations between parameters. The easiest approach is to assume that the parameters errors are independent and set the covariance between pairs of parameters equal to zero; \mathbf{B}_{pp} is then simply a diagonal matrix. Unfortunately, as we see in chapter 8, this assumption will not always hold. In situations where a model's parameters exhibit strong interdependence closer consideration will need to be given to the relationship between the parameters and non-zero cross covariances will need to be specified. In some situations it may be practical to transform the parameters to a new

set of uncorrelated variables.

6.3 Discussion

Advantages of the approximation (6.14) are that the matrix \mathbf{B}_k only needs to be updated at each new analysis time rather than at every time step and it does not require the previous cross covariance matrices to be stored. Since the number of parameters q is typically quite small, the computation of $\mathbf{N}_k\mathbf{B}_{pp}$ does not add significantly to the overall cost of the assimilation scheme. The block diagonal structure of (6.15) also makes computation of the matrix inverse \mathbf{B}_k^{-1} in the 3D-Var cost function more manageable. The reader is referred to appendix B for details of this inversion.

An increasingly popular approach to producing flow dependent forecast error covariances is to employ an Ensemble Kalman filter (EnKF) [40]. Although the EnKF method is cheaper and avoids many of the issues associated with the practical implementation of the traditional EKF it is still more expensive and less well established than 3D-Var. A notable problem with fully flow dependent methods such as the EKF and EnKF is filter divergence. This is where the filter becomes over confident in the accuracy of the analysis; the computed forecast error covariances become too small and consequently the analysis diverges from the true state. For the EnKF, this is likely to occur when the ensemble size is too small. Increasing the ensemble size can alleviate this problem but this will obviously have an impact on computational cost.

Assuming the prescribed state and parameter error variances do not underestimate the true error variances, we do not expect filter divergence to be an issue for our hybrid method. The matrices \mathbf{B}_{zz} and \mathbf{B}_{pp} are fixed and always non-zero; since \mathbf{B}_{pp} is fixed, the state-parameter cross covariances $\mathbf{B}_{zp_{k+1}} = \mathbf{N}_k\mathbf{B}_{pp}$ will only be zero when \mathbf{N}_k is zero. The Jacobian \mathbf{N}_k is essentially a measure of the sensitivity of the forecast model to the parameters; $\mathbf{N}_k = \mathbf{0}$ would imply that the model is insensitive to the parameters. In such a case the parameters will not be identifiable and any estimation process would be ineffectual.

There are a number of additional drawbacks to the EnKF, particularly when the dimen-

sion of the system is large compared to the number of ensemble members. The estimated covariance matrix is not full rank; the analysis increments are therefore restricted to the space spanned by the ensemble members. If the number of observations is considerably greater than the number of ensemble members this will lead to a significant loss of information as the background can only be corrected in a limited number of directions. This is known as the ‘rank problem’ and is discussed further in [40].

Another problem associated with using a relatively small ensemble to estimate a large covariance matrix is that spurious correlations can occur and the forecast error covariances will be non-smooth resulting in noisy updates. This can be treated by covariance localisation but this again introduces additional computational complexity to the scheme and can potentially lead to imbalance [41].

In our hybrid method, these problems are avoided simply by choosing an appropriate model for the state background error covariance matrix \mathbf{B}_{zz} . Depending on the method used to approximate \mathbf{N}_k , the cross covariances \mathbf{B}_{zp} could potentially become noisy but this will not directly affect the smoothness of the state updates.

6.4 Hybrid sequential data assimilation

For completeness we conclude this chapter with a summary of the hybrid matrix updating procedure for concurrent state-parameter estimation in the context of augmented sequential 3D-Var data assimilation.

Algorithm 6.1 The hybrid 3D-Var forecast-assimilation cycle

1. Starting from an initial augmented background state $\mathbf{w}_0^b = (\mathbf{z}_0^b, \mathbf{p}_0^b)^T$ containing estimates of the model state and parameters at t_0 , integrate the augmented system model (3.3) forward to the time t_k when a set of observations $\mathbf{y}_k = \mathbf{h}(\mathbf{z}_k) + \boldsymbol{\delta}_k$ first become available.
2. Update the state-parameter cross covariance matrix \mathbf{B}_{zp_k} by computing the Jacobian matrix \mathbf{N}_{k-1} (6.7). Combine with the fixed matrices \mathbf{B}_{zz} and \mathbf{B}_{pp} to form the augmented background error covariance matrix \mathbf{B}_k (6.15).

3. Minimise the augmented 3D-Var cost function (3.7) to obtain an augmented analysis vector $\mathbf{w}_k^a = (\mathbf{z}_k^a, \mathbf{p}_k^a)^T$.
4. Using the augmented model (3.3), evolve the augmented analysis forward to become the background estimate \mathbf{w}_{k+1}^b at time t_{k+1} . Note that the updated parameter values should be used in the state evolution model, i.e. $\mathbf{z}_{k+1}^b = \mathbf{f}(\mathbf{z}_k^a, \mathbf{p}_k^a)$.
5. If no observational data are available at time t_{k+1} set $\mathbf{w}_{k+1}^a = \mathbf{w}_{k+1}^b$.
6. Repeat steps 4. and 5. until a new set of observations are made then return to step 2.

6.5 Summary

In this chapter, we have proposed a new hybrid approach to modelling the augmented system background error covariances. By combining ideas from 3D-Var and the EKF we have developed a unique method that captures the flow dependent nature of the state-parameter cross covariances without the computational expense of explicitly propagating the full augmented covariance matrix \mathbf{B}_k . In the next chapter we explore the potential of the approach using a range of simple dynamical system models.

Chapter 7

Hybrid assimilation experiments

In this chapter we investigate the efficacy of our proposed new hybrid approach via a series of identical twin experiments with three simple general dynamical system models.

We start, in section 7.1, with the single parameter 1D linear advection equation introduced in chapter 5. In sections 7.2 and 7.3 we consider two simple general ODE models: a two parameter nonlinear damped oscillating system, and the three parameter nonlinear chaotic Lorenz 63 equations [60], [95]. In each section we give details of the specific model and its discretisation before using the method formulated in the previous chapter to derive estimates for the state-parameter cross covariances. A brief description of the experimental design is then given, followed by the results. We summarise our conclusions in section 7.4.

7.1 Linear advection

We first consider the single parameter one-dimensional linear advection equation introduced in section 5.1 [105],

$$\frac{\partial z}{\partial t} + c \frac{\partial z}{\partial x} = 0. \quad (7.1)$$

We assume that $z(x, t)$ is continuous and differentiable and choose to solve for $c > 0$ on a finite spatial domain with periodic boundary conditions. In chapter 5 we solved

(7.1) using a semi-Lagrangian method. For the experiments presented in this section we rescale our model and discretise using the upwind scheme (e.g. [53], [71])

$$z_{k+1,j} = z_{k,j} - c \frac{\Delta t}{\Delta x} (z_{k,j} - z_{k,j-1}), \quad j = 1, 2, \dots, J \quad k = 0, 1, \dots, T \quad (7.2)$$

with boundary conditions

$$z_{k,0} = z_{k,J} \quad (7.3)$$

where $z_{k,j} \approx z(x_j, t_k)$ and $x_j = j\Delta x$, $t_k = k\Delta t$, and Δx is the spatial grid spacing and Δt is the model time step.

Denoting $\mu = \frac{\Delta t}{\Delta x}$, we can rewrite (7.2) as

$$z_{k+1,j} = (1 - c\mu)z_{k,j} + c\mu z_{k,j-1}. \quad (7.4)$$

The upwind scheme is first order accurate in space and time and stable provided that the CFL condition $c\mu \leq 1$ is satisfied. To ensure that the model remains stable during the assimilation we set $\mu = 1$ and assume that c is known to be somewhere on the interval $0 \leq c \leq 1$. The upwind scheme is numerically diffusive; this results mainly in amplitude errors in the solution when the model is run with the correct c value and without any data assimilation. Full details of the model validation are given in appendix C. The reason for using the discretisation (7.4) is that it can be expressed as a linear matrix system. This will be advantageous when computing the Jacobian matrix \mathbf{N}_k .

The state forecast model (7.4), with known constant advection speed c , can be written as

$$\mathbf{z}_{k+1} = \mathbf{M}\mathbf{z}_k \quad (7.5)$$

where $\mathbf{z}_k \in \mathbb{R}^m$ is the model state at time t_k and \mathbf{M} is a (constant) $m \times m$ matrix, that depends nonlinearly on the advection velocity c ,

$$\mathbf{M} = \mathbf{M}(c) = \begin{pmatrix} (1 - c\mu) & 0 & & & c\mu \\ c\mu & (1 - c\mu) & 0 & \dots & \\ & \ddots & \ddots & \ddots & \\ & & & & 0 \\ 0 & \dots & 0 & c\mu & (1 - c\mu) \end{pmatrix}. \quad (7.6)$$

Since the advection velocity c is constant, the parameter evolution model (3.1) is given by

$$c_{k+1} = c_k. \quad (7.7)$$

Setting

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{z}_k \\ c_k \end{pmatrix}, \quad (7.8)$$

we can combine (7.5) and (7.7) to give the augmented system model

$$\mathbf{w}_{k+1} = \tilde{\mathbf{f}}(\mathbf{w}_k) \quad (7.9)$$

$$= \begin{pmatrix} \mathbf{M}_k & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{z}_k \\ c_k \end{pmatrix} \quad (7.10)$$

Note that the constant matrix \mathbf{M} in (7.5) has been replaced by the time varying matrix $\mathbf{M}_k = \mathbf{M}(c_k)$. Although the true matrix \mathbf{M} is constant, the forecast model at time t_k will depend on the current estimate, c_k , of the true advection velocity, c . The matrix \mathbf{M}_k will therefore vary as c_k is updated by the assimilation process.

7.1.1 State-parameter cross covariance

For the approximation of the cross covariances between the errors in the model state \mathbf{z} and parameter c at time t_{k+1} we need to calculate the matrix \mathbf{N}_k ; the Jacobian of the state forecast model with respect to the model parameters. For the linear advection model this is given by

$$\mathbf{N}_k = \left. \frac{\partial (\mathbf{M}_k \mathbf{z}_k)}{\partial c} \right|_{\mathbf{z}_k^a, c_k^a}. \quad (7.11)$$

From (7.6) we have

$$\frac{\partial \mathbf{M}_k}{\partial c_k} = \begin{pmatrix} -\mu & 0 & & \mu \\ \mu & -\mu & 0 & \dots \\ & \ddots & \ddots & \ddots \\ & & & 0 \\ 0 & \dots & 0 & \mu & -\mu \end{pmatrix}. \quad (7.12)$$

which is constant for all c_k .

The matrix \mathbf{N}_k is a $m \times 1$ vector with elements N_{k_j}

$$N_{k_j} = -\mu(z_{k,j} - z_{k,j-1}), \quad j = 1, \dots, m. \quad (7.13)$$

Multiplying (7.13) with the parameter error variance σ_c^2 gives the matrix $\mathbf{B}_{z\mathbf{p}_{k+1}}$ elements as

$$b_{z\mathbf{p}_{k+1},j} = -\sigma_c^2 \mu(z_{k,j} - z_{k,j-1}), \quad j = 1, \dots, m. \quad (7.14)$$

where $b_{z\mathbf{p}_{k+1},j}$ represents the cross covariance between the parameter error $\varepsilon_{c_{k+1}}$ and element j of the state background error vector $\varepsilon_{z_{k+1}}^b$ at time t_{k+1} .

We note that if we take

$$\frac{z_{k,j} - z_{k,j-1}}{\Delta x} \approx \tilde{f}'_k(x_j - ct_k) \quad (7.15)$$

to be an approximation to the gradient of the true solution at time t_k , we can write (7.14) as

$$b_{z\mathbf{p}_{k+1},j} = -\sigma_c^2 \Delta t \tilde{f}'_k(x_j - ct_k), \quad j = 1, \dots, m. \quad (7.16)$$

If we compare (7.16) with the approximation (5.43) derived in chapter 5, section 5.6 we find that the two expressions are very similar, the main difference being that (7.16) uses a new estimate of the gradient \tilde{f}'_k at each assimilation time. The form (7.14) is used in the experiments described in the next section.

7.1.2 Assimilation experiments

We run the linear advection model on the domain $x \in [0, 3]$ with grid spacing $\Delta x = 0.01$ and time step $\Delta t = 0.01$. The true initial data is given by the Gaussian function

$$z(x, 0) = \begin{cases} 0 & x < 0.01 \\ e^{-\frac{(x-0.25)^2}{0.01}} & 0.01 < x < 0.5 \\ 0 & x \geq 0.5 \end{cases} \quad (7.17)$$

The true advection velocity is set as $c = 0.5$ and the initial model estimate for c is generated by adding Gaussian random noise with zero mean and variance $\sigma_c^2 = 0.1$ to this value. This corresponds to an error variance of 20%.

The initial model background state for z is also a Gaussian but is rescaled so that it is a different height, width and in a slightly different starting position to the truth. The state background error covariance matrix \mathbf{B}_{zz} is defined as in section 5 equation (5.23) with background error variance $\sigma_b^2 = 0.05$. The correlation length scale L in (5.23) is set at twice the current observation spacing. The parameter error covariance matrix \mathbf{B}_{pp} is simply the scalar quantity $\sigma_c^2 = 0.1$.

The analytic solution (5.5) is used to provide observations for the assimilation and also to evaluate the performance of the scheme in terms of estimating the model state z . The observations are generated by sampling the analytic solution on a regularly spaced grid and are initially assumed to be perfect. The spatial and temporal frequency of the observations remains fixed for the duration of each individual assimilation experiment but is varied between experiments as described below in section 7.1.3. The observation operator therefore takes the form

$$\tilde{\mathbf{H}}_k = \tilde{\mathbf{H}} \equiv \begin{pmatrix} \mathbf{H} & \mathbf{0} \end{pmatrix} \quad \text{for all } k. \quad (7.18)$$

where $\mathbf{H} \in \mathbb{R}^{r \times m}$, with the number of observations $r_k = r$ the same for all k .

We assume that the observation errors are spatially and temporally uncorrelated and set the observation error covariance matrix \mathbf{R}_k

$$\mathbf{R}_k = \mathbf{R} = \sigma_o^2 \mathbf{I}, \quad \mathbf{I} \in \mathbb{R}^{r \times r}, \quad (7.19)$$

where σ_o^2 is the observation error variance.

The assimilation process is carried out sequentially as described in algorithm 6.1. The augmented cost function is minimised iteratively using the quasi-Newton descent algorithm described in section 5.2.

7.1.3 Results

Perfect observations

Experiments were carried out using a range of both over- and under-estimated initial c values and different initial background guesses for \mathbf{z}^b . We found that the quality of the state analysis and convergence and accuracy of the parameter estimates depends on a number of factors including the initial background guess, the location and spatial frequency of the observations, the time between successive assimilations, and the presence of observational noise. Here, we discuss the results from one example case; the advection velocity c is initially over estimated as $c_0 = 0.87116$ and the initial model background bathymetry is slightly shorter, narrower and positioned behind the true $z(x, 0)$ as illustrated in the first panel of figure 7.1.

Figure 7.2 shows the effect on the parameter c estimates of varying the spatial frequency of the observations. Observations are assimilated every $10\Delta t$ with observation error variance $\sigma_o^2 = 0.01$. The grid spacing between observations ranges from every Δx to every $50\Delta x$. The hybrid approach works extremely well. For observations taken at intervals between Δx and $25\Delta x$ the true value of c is found to a high level of accuracy. The speed of convergence decreases as the number of observations decreases. For intervals of Δx , $5\Delta x$ and $10\Delta x$ the analysis for z is consistently good and closely tracks the true solution. As figure 7.1 illustrates, when observations are taken every $25\Delta x$ we find that the model does not always correctly predict the true height of the curve. If the observation spacing is further increased to $50\Delta x$, there is much greater variation in the parameter estimates. The estimates get close to but never quite settle on the true value of c , even when the assimilation period is extended beyond that shown. The corresponding state estimates are shown in figure 7.4. Although the predicted model state remains approximately in phase with the true solution, the observations are too widely spaced to be able to accurately reproduce the characteristics of the curve.

Figure 7.3 shows the effect of varying the temporal frequency of the observations between $5\Delta t$ and $50\Delta t$. For these experiments, the spatial frequency is fixed at $10\Delta x$. The results are similar to the previous experiment; the speed of convergence decreases as the frequency of the observations decrease but the final estimated c values are very close to

the true value. There are only small differences in convergence when the time between successive assimilations is increased from every $5\Delta t$ to $10\Delta t$ to $25\Delta t$. The analysis is for z is also very good, with only slight fluctuations in height for the case $25\Delta t$. When observations are taken every $50\Delta t$ the c estimate takes much longer to converge and as a result the model takes longer to stabilise. As can be seen in figure 7.5, once the model has settled the analysis for z is relatively good. Although again we see a similar problem as in figure 7.1 above with the true height of the curve being underestimated at times when the peak of the curve is not directly observed. If the time between observations is doubled to $100\Delta t$ the scheme completely fails to recover c .

Noisy observations

The effect of observational errors was investigated by adding random noise to the observations. This noise was defined to have a Gaussian distribution with mean zero and variance σ_o^2 where σ_o^2 is the observation error variance. Observations were taken at spatial intervals of $10\Delta x$ and assimilated every $10\Delta t$. The top panel in figure 7.6 shows the parameter c estimates produced for observation error variance increasing from $\sigma_o^2 = 0.001$ to $\sigma_o^2 = 0.1$. This represents errors with variance of up to 10% of the maximum curve height.

As we would expect, when the observations are noisy the resulting state analysis and parameter estimates are also noisy. The amplitude of oscillations in c increase as σ_o^2 is increased. The oscillations are, however, approximately centered around the true c value and lie within the bounds of uncertainty placed on the observations. We found that smoother and more accurate parameter estimates could be obtained by averaging over a moving time window as the assimilation is running as is shown in the bottom panel of figure 7.6. Here, the above experiments were repeated but with the c estimates being averaged over a moving time window of 40 time steps. Note that to allow time for the scheme to settle we omit the early estimates and begin the averaging at $t = 2.0$.

When $\sigma_o^2 = 0.001$ the impact of the noise on the analysis for z is relatively minor. When σ_o^2 is increased the effect of the noise becomes more noticeable. As figure 7.7 shows, when $\sigma_o^2 = 0.01$, the main body of the curve is reproduced to a high level of accuracy but oscillations appear in the tails of the curve. When the observation error variance

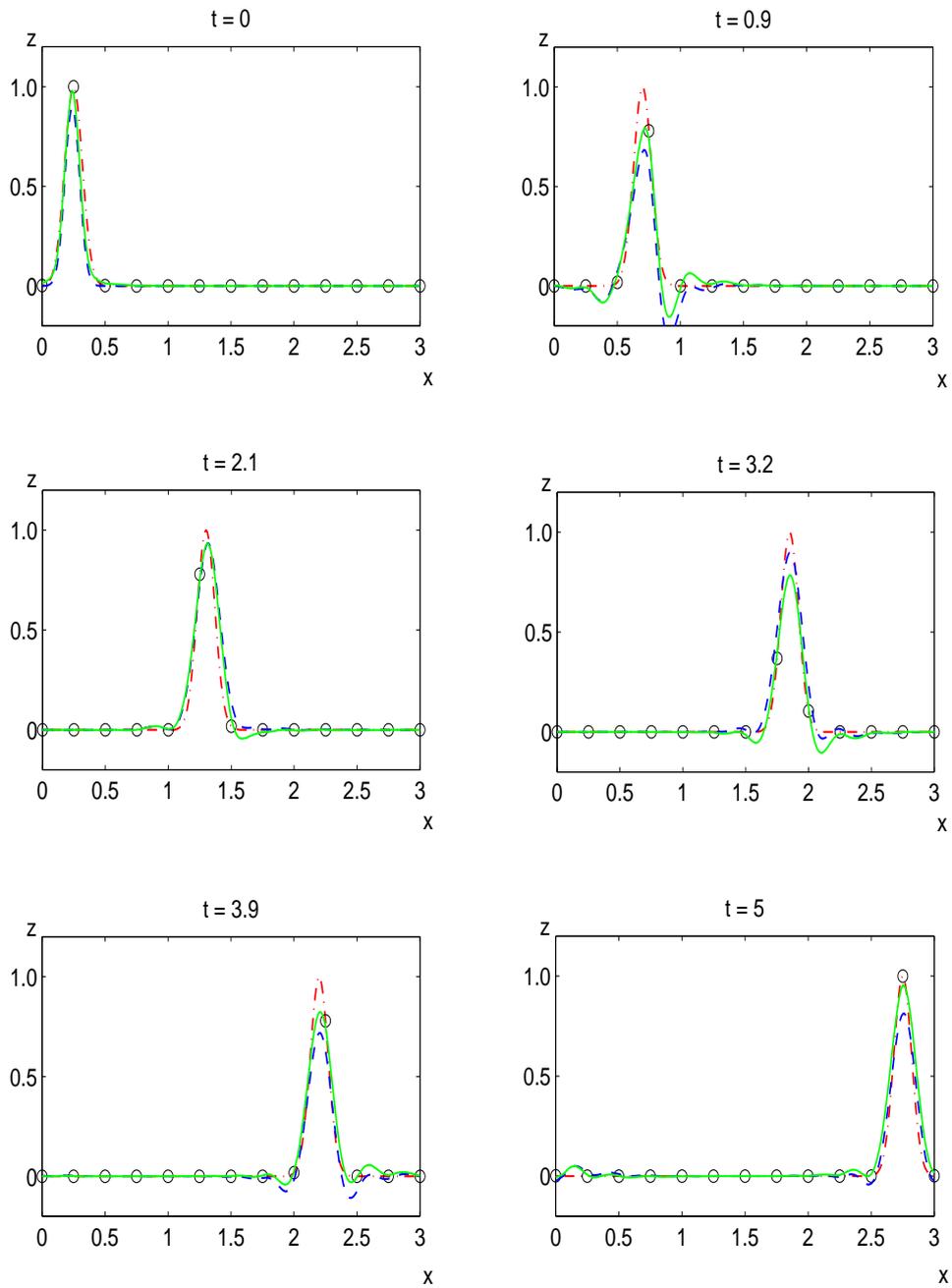


Figure 7.1: Linear advection: state analysis at times $t = 0, 0.9, 2.1, 3.2, 3.9$ & 5 , with perfect observations every $25\Delta x$, $10\Delta t$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

is further increased to $\sigma_o^2 = 0.1$ the oscillations become much larger (figure 7.8). The noise now pollutes the entire solution, but the model does move roughly in phase with the true solution.

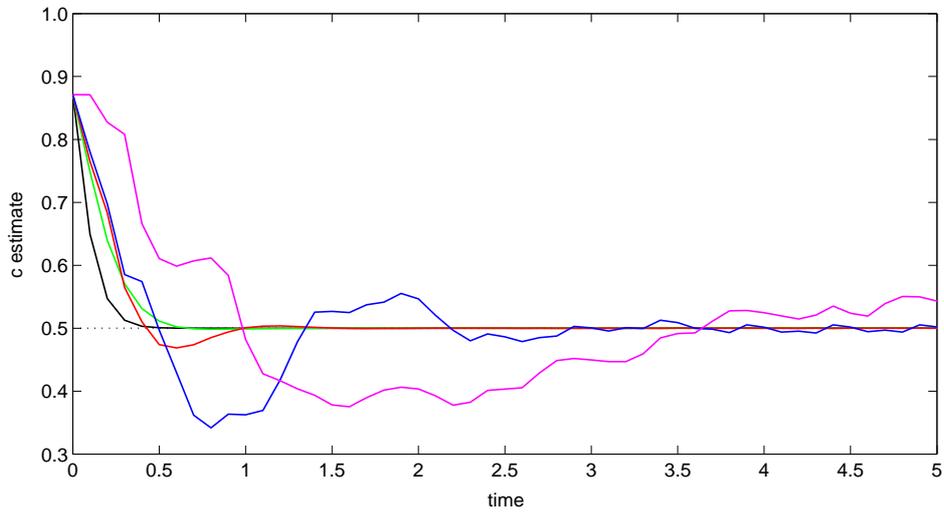


Figure 7.2: Linear advection: varying the spatial frequency of observations. Parameter updates for initial estimate $c = 0.87116$: solid black line - observations at $1\Delta x$ intervals; solid green line - observations at $5\Delta x$ intervals; solid red line - observations at $10\Delta x$ intervals; solid blue line - observations at $25\Delta x$ intervals; solid pink line - observations at $50\Delta x$ intervals. The dotted black line indicates the true parameter c value.

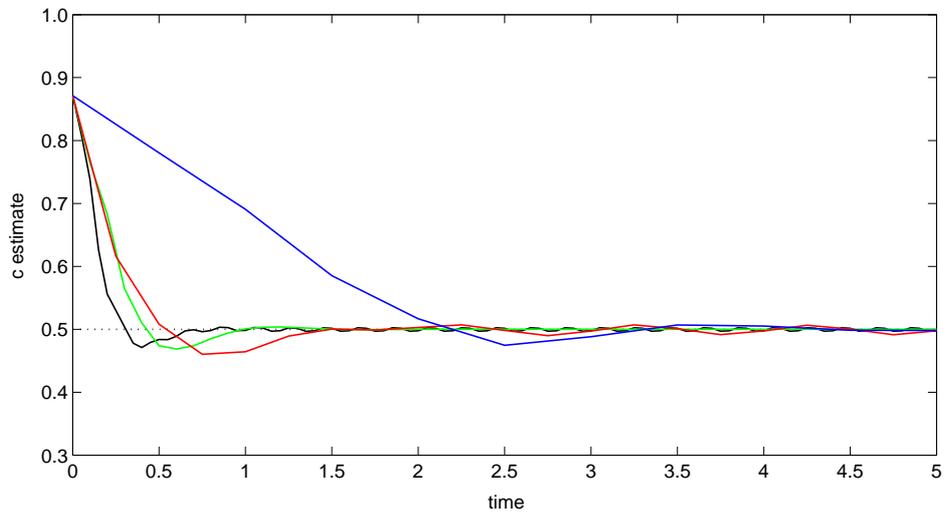


Figure 7.3: Linear advection: varying the temporal frequency of observations. Parameter updates for initial estimate $c = 0.87116$: solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $25\Delta t$; solid blue line - observations every $50\Delta t$. The dotted black line indicates the true parameter c value.

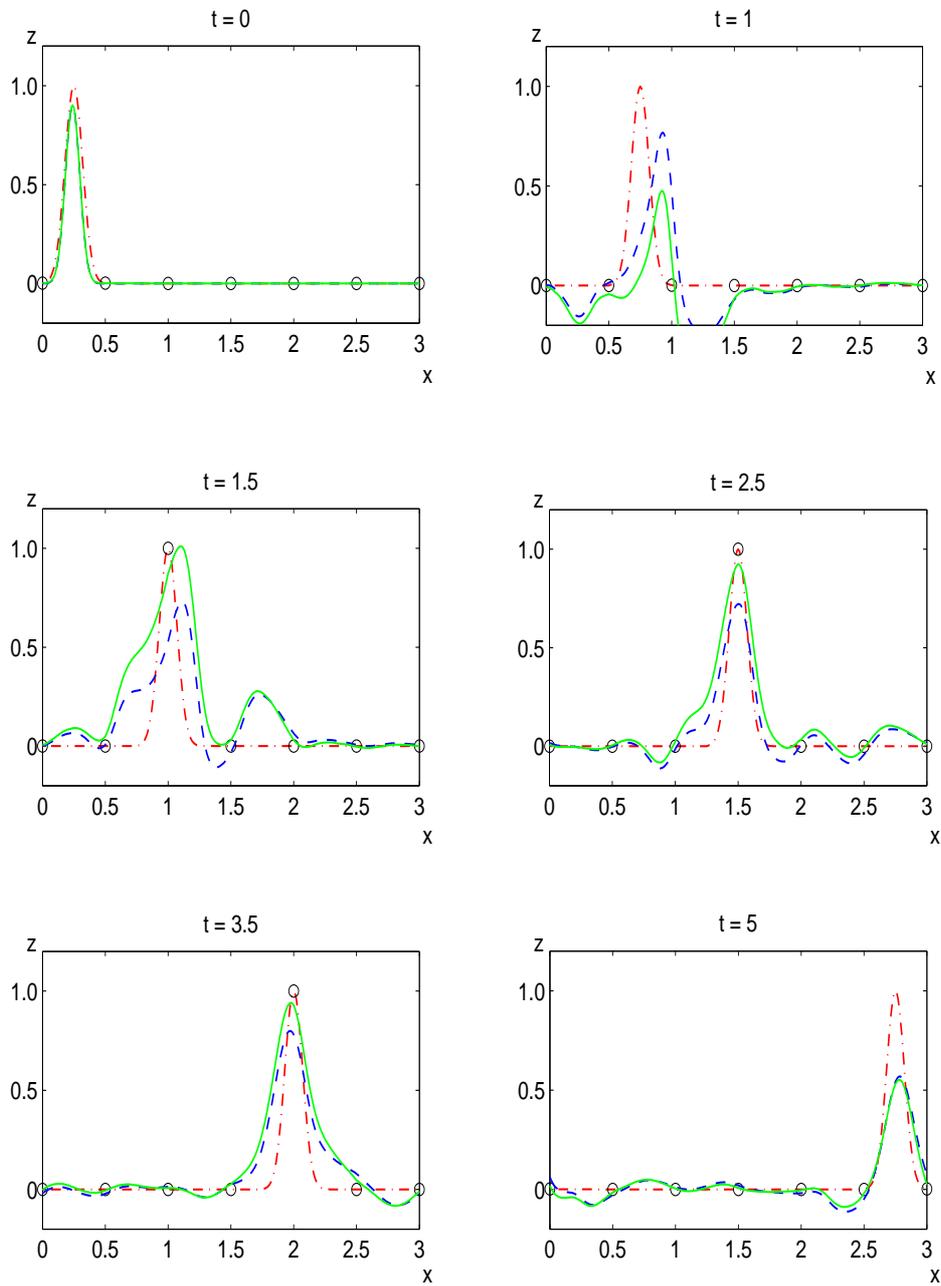


Figure 7.4: Linear advection: state analysis at times $t = 0, 1, 1.5, 2.5, 3.5$ & 5 , with perfect observations every $50\Delta x$, $10\Delta t$. The red dot-dash line represents the true solution z^t , observations y are given by circles, the background z^b is given by the dashed blue line and the analysis z^a is given by the solid green line.

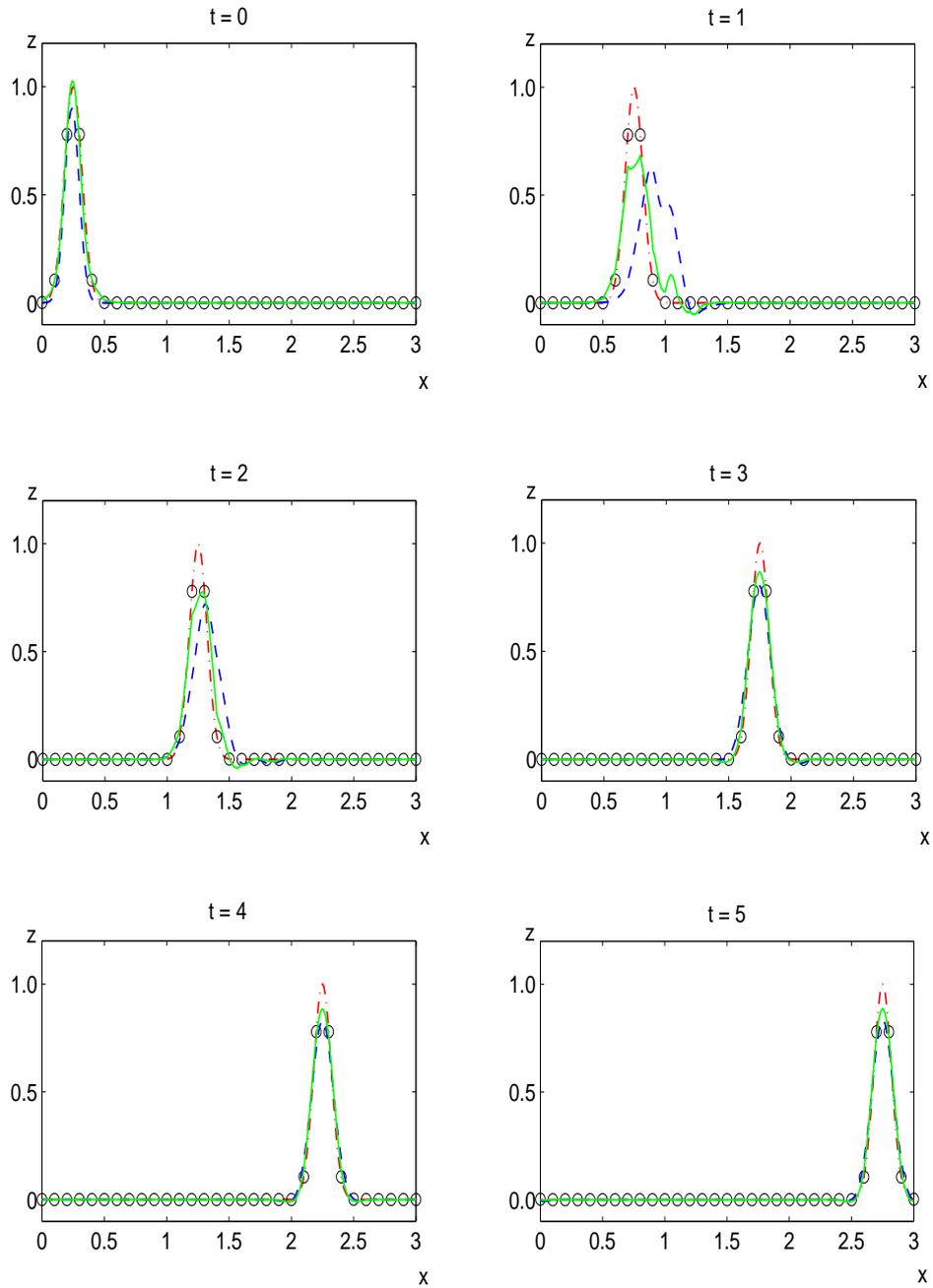


Figure 7.5: Linear advection: state analysis at times $t = 0, 1, 2, 3, 4$ & 5 , with perfect observations every $50\Delta t$, $10\Delta x$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

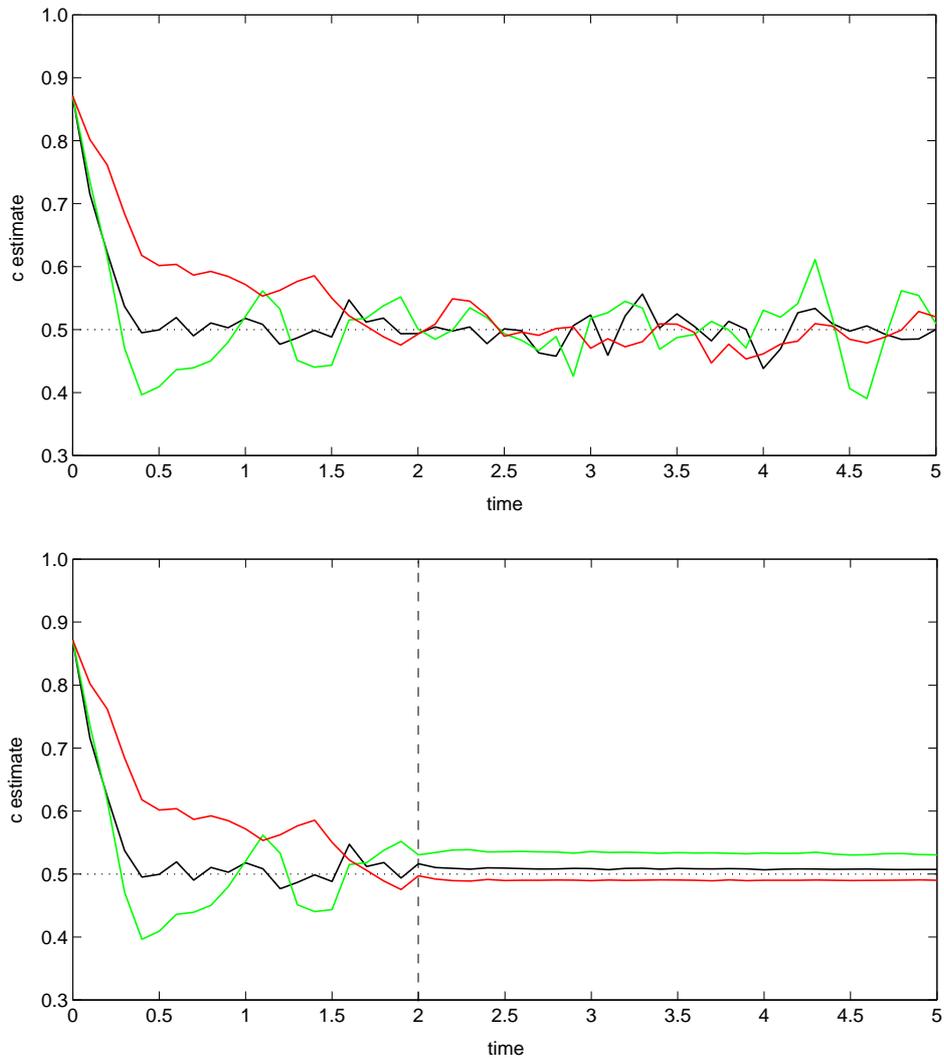


Figure 7.6: Linear advection model: imperfect observations. Parameter updates for initial estimate $c = 0.87116$. **Top:** unaveraged estimates. **Bottom:** time averaged estimates. Solid black line - $\sigma_o^2 = 0.001$; solid green line - $\sigma_o^2 = 0.01$; solid red line - $\sigma_o^2 = 0.1$. The dashed black line indicates the start of parameter averaging. The dotted black line indicates the true parameter c value.

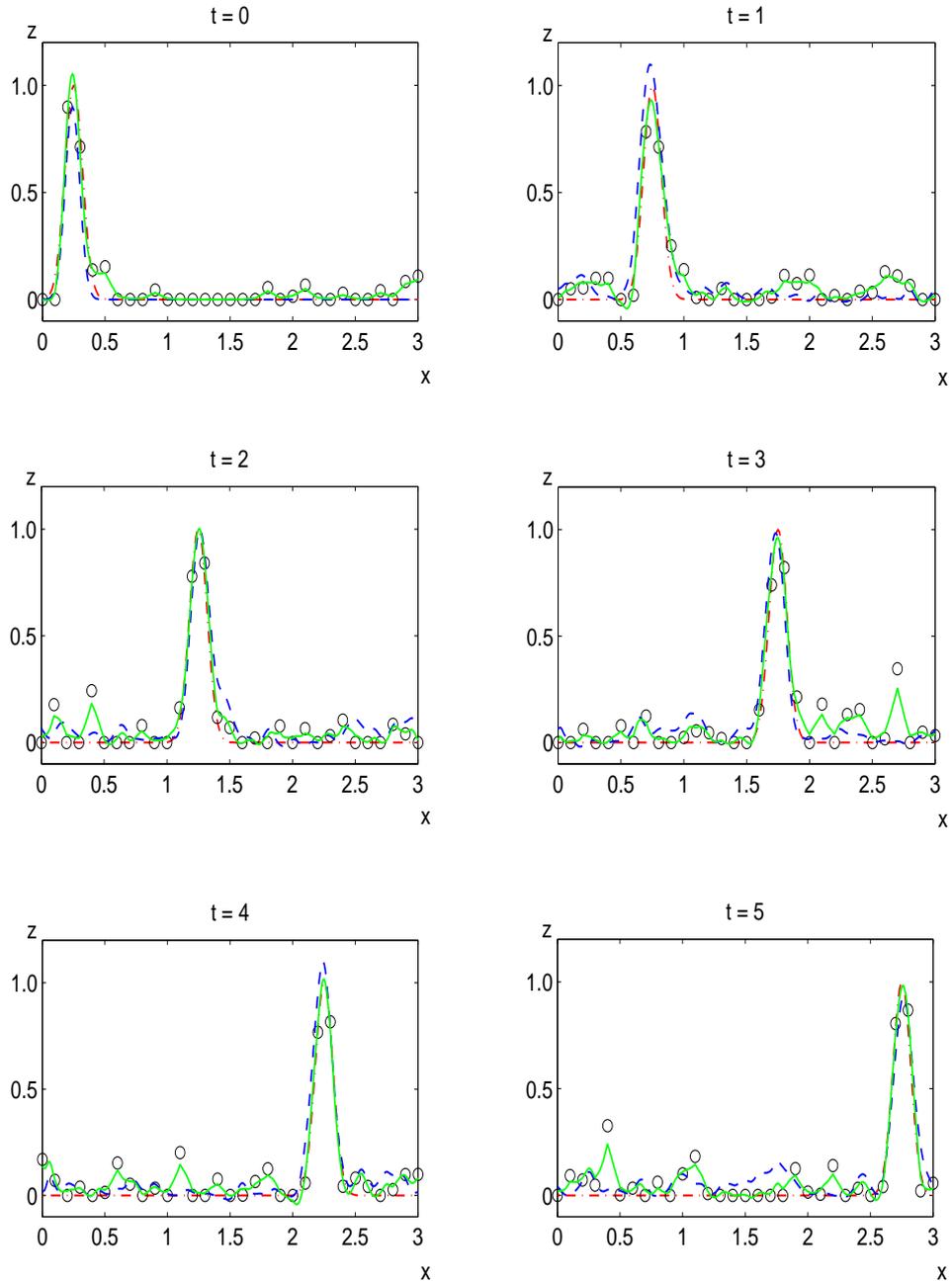


Figure 7.7: Linear advection model: state analysis at times $t = 0, 1, 2, 3, 4$ & 5 , with noisy observations every $10\Delta x$, $10\Delta t$ and $\sigma_o^2 = 0.01$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

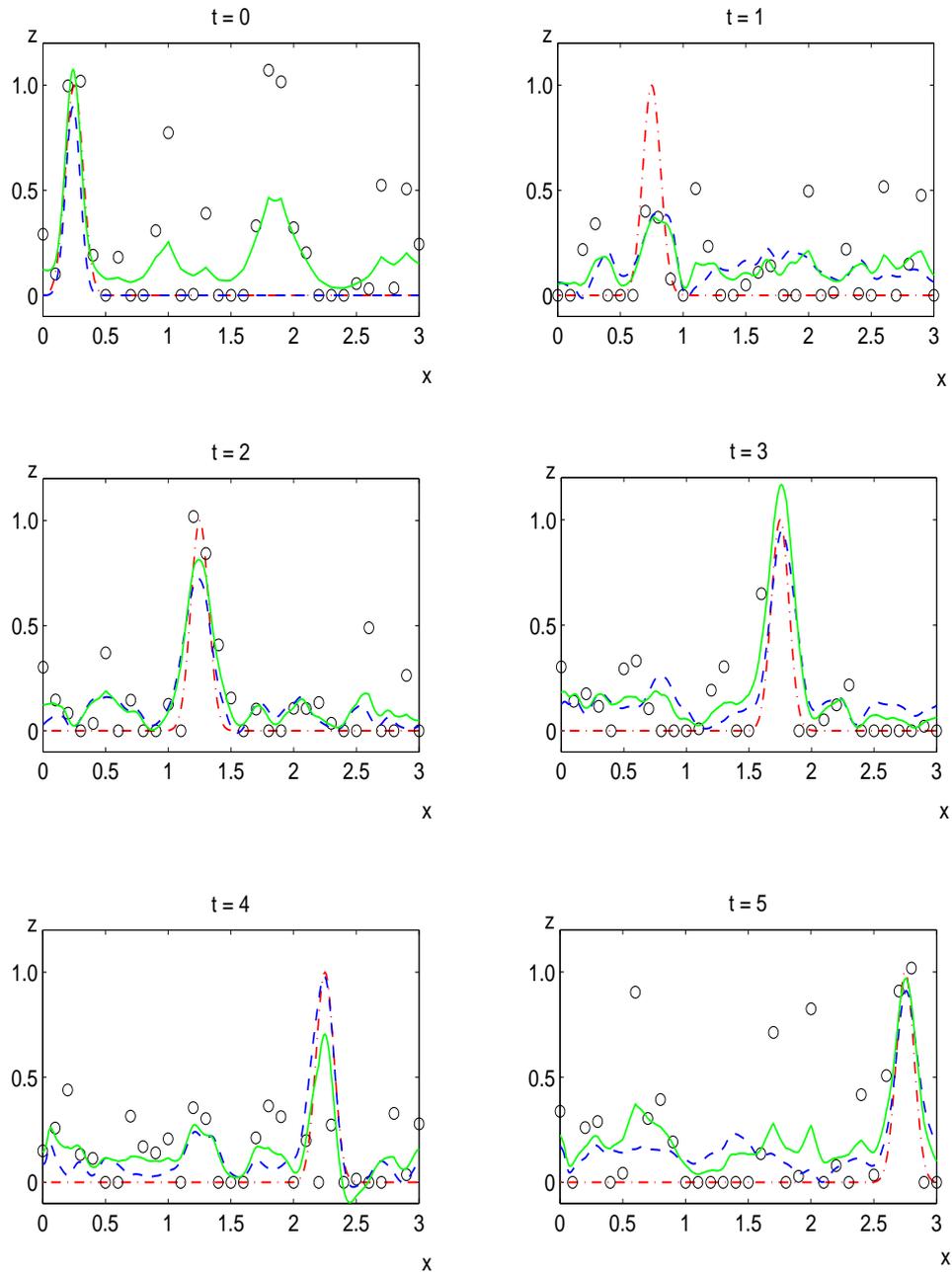


Figure 7.8: Linear advection model: state analysis at times $t = 0, 1, 2, 3, 4$ & 5 , with noisy observations every $10\Delta x$, $10\Delta t$ and $\sigma_o^2 = 0.1$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

7.2 Nonlinear oscillator

Our second test model is a two parameter, unforced, damped nonlinear oscillator given by the second order ordinary differential equation

$$\ddot{x} + d\dot{x} + mx + x^3 = 0. \quad (7.20)$$

where d and m are real, constant, parameters and $x = x(t)$.

Equation (7.20) is often referred to as the Duffing equation or Duffing oscillator. It arises in a variety of applications and in a number of different forms. For a more detailed discussion see e.g. [35], [106], [99]. For $d, m > 0$, the form (7.20) describes the motion of a single mass attached to a spring with nonlinear elasticity and linear damping. The parameter d is the damping coefficient and m is the square of the frequency of oscillation. The quantity $-(mx + x^3)$ is known as the restoring or spring force and represents the force exerted by the spring when it is subjected to a displacement x [98].

Equation (7.20) can be written as the first order system

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= -(mx + x^3 + dy). \end{aligned} \quad (7.21)$$

The nature of the solution of this system varies greatly depending on the values of the parameters. For $d, m > 0$ the system (7.21) has a single stable equilibrium at $(x, \dot{x}) = (0, 0)$.

We solve (7.21) numerically using a second order Runge-Kutta method (see e.g. [11]). The discrete system is given by the following set of equations

$$x_{k+1} = \left(\Delta t - d \frac{\Delta t^2}{2} \right) y_k + \left(1 - m \frac{\Delta t^2}{2} - \frac{\Delta t^2}{2} x_k^2 \right) x_k \quad (7.22)$$

$$\begin{aligned} y_{k+1} &= \left(1 - d\Delta t - m \frac{\Delta t^2}{2} + d^2 \frac{\Delta t^2}{2} \right) y_k + \left(-m\Delta t + dm \frac{\Delta t^2}{2} + \left(d \frac{\Delta t^2}{2} - \frac{\Delta t}{2} \right) x_k^2 \right) x_k \\ &\quad - \frac{\Delta t}{2} (x_k + \Delta t y_k)^3. \end{aligned} \quad (7.23)$$

We combine the model parameters d and m in the parameter vector $\mathbf{p}_k \in \mathbb{R}^2$

$$\mathbf{p}_k = \begin{pmatrix} d_k \\ m_k \end{pmatrix}. \quad (7.24)$$

with

$$\mathbf{p}_{k+1} = \mathbf{p}_k, \quad k = 0, 1, \dots, T. \quad (7.25)$$

Adding the parameter vector \mathbf{p}_k to the state vector

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ y_k \end{pmatrix}, \quad (7.26)$$

gives the augmented state vector

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{p}_k \end{pmatrix} = \begin{pmatrix} x_k \\ y_k \\ d_k \\ m_k \end{pmatrix}. \quad (7.27)$$

This allows us to write (7.22)-(7.23) and (7.25) as the equivalent augmented system

$$\begin{aligned} \mathbf{w}_{k+1} &= \tilde{\mathbf{f}}(\mathbf{w}_k) \\ &= \begin{pmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix}. \end{aligned} \quad (7.28)$$

where $\mathbf{f} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ and $\tilde{\mathbf{f}} : \mathbb{R}^4 \rightarrow \mathbb{R}^4$.

7.2.1 State-parameter cross covariance

For the oscillating system, the Jacobian of the state forecast model with respect to the parameters is a 2×2 matrix defined as

$$\begin{aligned} \mathbf{N}_k &= \left(\begin{array}{cc} \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial d} & \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial m} \end{array} \right) \Big|_{\mathbf{x}_k^a, \mathbf{p}_k^a} \\ &= \left(\begin{array}{cc} \frac{\partial x_{k+1}}{\partial d} & \frac{\partial x_{k+1}}{\partial m} \\ \frac{\partial y_{k+1}}{\partial d} & \frac{\partial y_{k+1}}{\partial m} \end{array} \right) \Big|_{d_k, m_k} \end{aligned} \quad (7.29)$$

The elements of \mathbf{N}_k can be computed directly from the discrete equations (7.22)-(7.23).

Details of this calculation are given in appendix D.

We assume that the parameters d and m are uncorrelated and set the parameter error covariance matrix

$$\mathbf{B}_{\mathbf{pp}} = \begin{pmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_m^2 \end{pmatrix}, \quad (7.30)$$

where σ_d^2 and σ_m^2 are the error variances of the parameters d and m respectively.

The state-parameter cross covariance matrix is then given by

$$\begin{aligned} \mathbf{B}_{\mathbf{x}\mathbf{p}_{k+1}} &= \mathbf{N}_k \mathbf{B}_{\mathbf{pp}} \\ &= \left(\begin{array}{cc} \sigma_d^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial d} & \sigma_m^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial m} \end{array} \right) \Big|_{\mathbf{x}_k^a, \mathbf{p}_k^a}. \end{aligned} \quad (7.31)$$

7.2.2 Assimilation experiments

The scheme is tested using identical twin experiments. We define the ‘true’ solution to be that given by the discretised equations (7.22) - (7.23) with model time step $\Delta t = 0.1$, initial displacement $x_0 = 2.0$, initial velocity $y_0 = 0.0$ and parameter values $d = 0.05$ and $m = 1.0$. The solution for x and y is shown in figure 7.9 for $t \in [0, 50]$.

The initial background estimate for the state \mathbf{x}_0^b is generated by adding random noise to the the true initial conditions. This noise is taken from a Gaussian distribution with zero mean and variance $\sigma_b^2 = 0.01$. The state background error covariance matrix is set as the diagonal matrix

$$\mathbf{B}_{\mathbf{xx}} = \sigma_b^2 \mathbf{I}, \quad \mathbf{I} \in \mathbb{R}^{2 \times 2}. \quad (7.32)$$

Observations of both x and y are taken from the true model trajectory and are assimilated at regular time intervals. The augmented observation operator is simply

$$\tilde{\mathbf{H}} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \end{pmatrix} \quad \text{for all } k, \quad (7.33)$$

with $\mathbf{I} \in \mathbb{R}^{2 \times 2}$.

For the observation error covariance matrix we use

$$\mathbf{R} = \sigma_o^2 \mathbf{I}, \quad \mathbf{I} \in \mathbb{R}^{2 \times 2}. \quad (7.34)$$

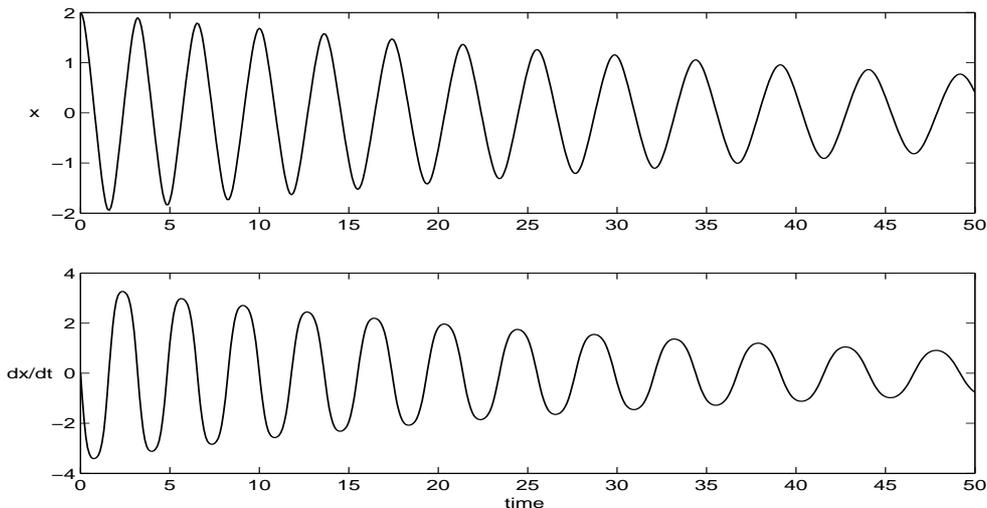


Figure 7.9: Damped, unforced nonlinear oscillator: computed numerical solution for x and $y = \frac{dx}{dt}$.

We generate initial estimates d_0 and m_0 for the parameters d and m by adding random noise with error variance $\sigma_d^2 = 0.005$ and $\sigma_m^2 = 0.1$ respectively to the ‘true’ values. It is assumed that we know d and m are positive. Since the dimension of augmented state is small, we compute the analysis directly from the augmented OI analysis equation (3.12).

7.2.3 Results

Perfect observations

Figures 7.10 and 7.11 show the parameter d and m updates for two example model runs with initial parameter estimates (i) $d_0 = 0.081877$ (over estimated), $m_0 = 0.58617$ (under estimated) and (ii) $d_0 = 0.0193$ (under estimated), $m_0 = 1.4711$ (over estimated), with observations of x and $y = \frac{dx}{dt}$ at varying temporal frequencies. The observation error variance is set at $\sigma_o^2 = 0.01$.

In the first example (figure 7.10) the scheme manages to retrieve the true values of both d and m to a good level of accuracy for observation intervals up to every $25\Delta t$. We see a large increase in error in the estimated parameter values when the observation frequency is decreased to $50\Delta t$. This is reflected in the analysis for x and y and can be seen by comparing figures 7.12 and 7.13. The x and y analyses for observation intervals of less than $25\Delta t$ are not shown as qualitatively they are indistinguishable from the true

solution. When observations are taken every $25\Delta t$ we start to see very small differences (figure 7.12), and when the observation frequency is halved to $50\Delta t$ (figure 7.13) there is a marked deterioration in the analysis. The over estimation of both d and m causes significant phase and amplitude errors. We can see from figure 7.13 that too much weight is being given to the background state and suggests that our background error variance estimate of $\sigma_b^2 = 0.01$ is too low in this case. Indeed, we would expect the errors in the model forecast to grow as the time between successive assimilations is increased. Figure 7.14 and the dashed pink line in figures 7.12 (a) and (b) show the results produced when the same experiment was repeated but with the background error variance doubled to $\sigma_b^2 = 0.02$. More weight is now given to the observations, the error in the parameter d and m estimates is considerably reduced and hence the quality of the state analysis is greatly improved.

For example (ii) (figure 7.11), the scheme produces accurate estimates of both d and m for observation frequencies Δt , $5\Delta t$ and $10\Delta t$ but the results are poor for observation intervals of $25\Delta t$ and $50\Delta t$. The estimate of the parameter d for $25\Delta t$ is around twice its true value. The quality of analysis for x and y is also similar to that in figure 7.13 and so is not shown. When observations are assimilated every $50\Delta t$ the d and m estimates are beyond any reasonable error bounds. As in example (i), this has a detrimental effect on the model and therefore the state analysis. However, we again found that we could significantly improve both the state and parameter estimates by inflating σ_b^2 (not shown).

Noisy observations

The assimilation experiments were repeated using imperfect observations. Figures 7.15 (a) and (b) show the results produced when random noise with variance $\sigma_o^2 = 0.01$ was added to the observations as described in section 7.1.3 above. Note that the assimilation time has been increased. Overall the scheme performs well, although there is some interesting behaviour. Unlike the perfect observation case, there is no clear relationship between the observation frequency and the accuracy of the parameter estimates; the best estimates of d and m were obtained using different observation intervals. There was also variation across model runs. The most noticeable result is the parameter d

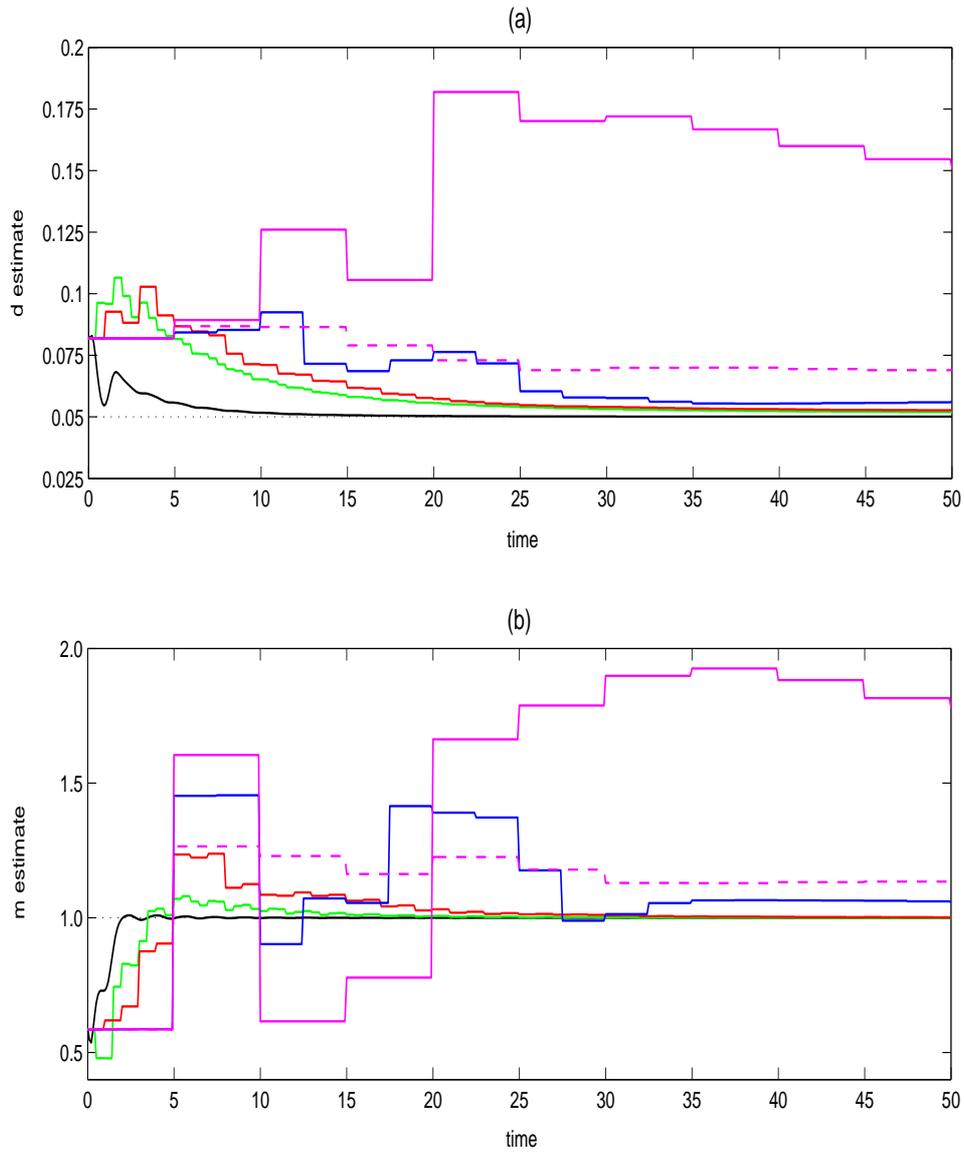


Figure 7.10: Nonlinear oscillator: **(a)** Parameter d updates for initial estimate $d_0 = 0.081877$, **(b)** Parameter m updates for initial estimate $m_0 = 0.58617$: solid black line - observations every Δt ; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid pink line - observations every $50\Delta t$; dashed pink line - observations every $50\Delta t$ with inflated σ_b^2 . The dotted black line indicates the true parameter value.

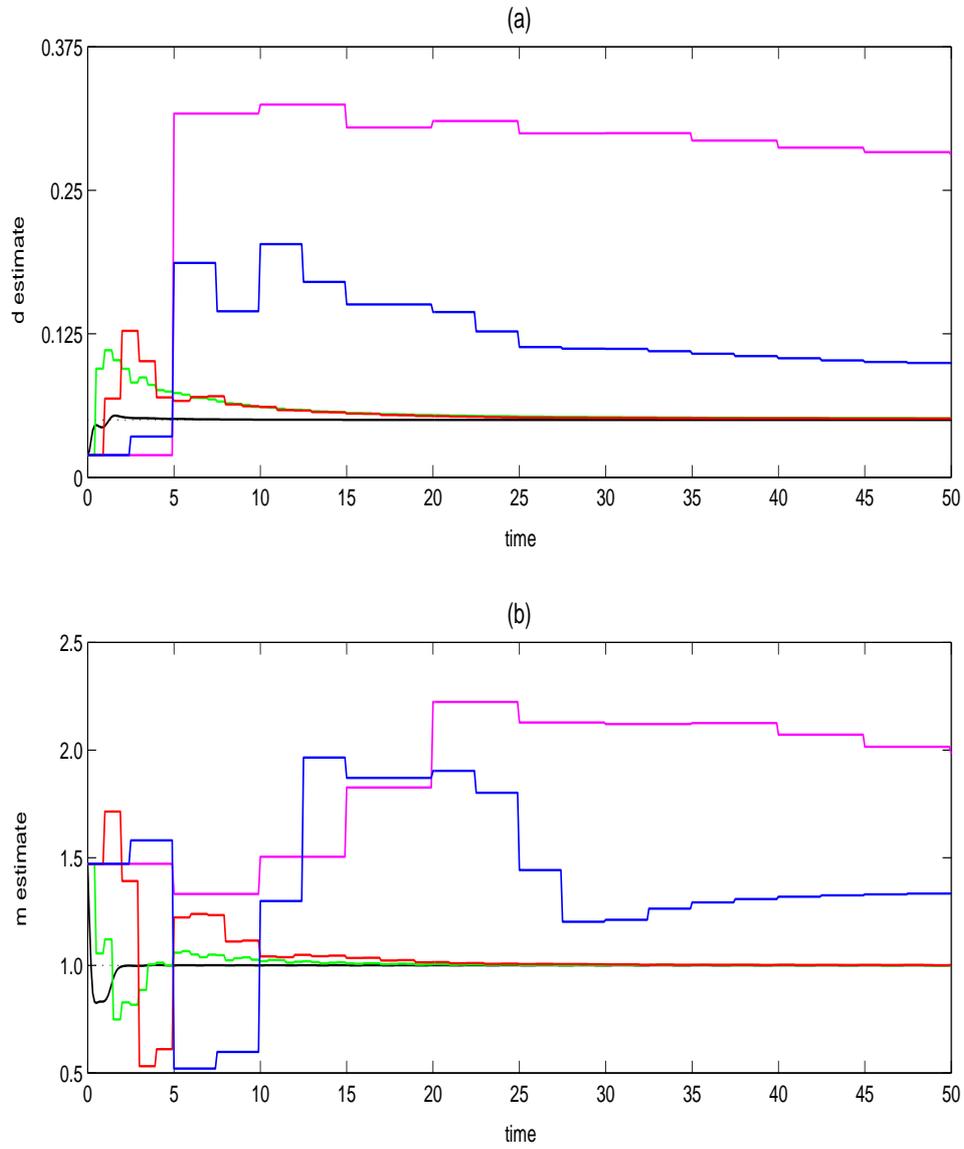


Figure 7.11: Nonlinear oscillator **(a)** Parameter d updates for initial estimate $d_0 = 0.0193$, **(b)** Parameter m updates for initial estimate $m_0 = 1.4711$: solid black line - observations every Δt ; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid pink line - observations every $50\Delta t$. The dotted black line indicates the true parameter value.

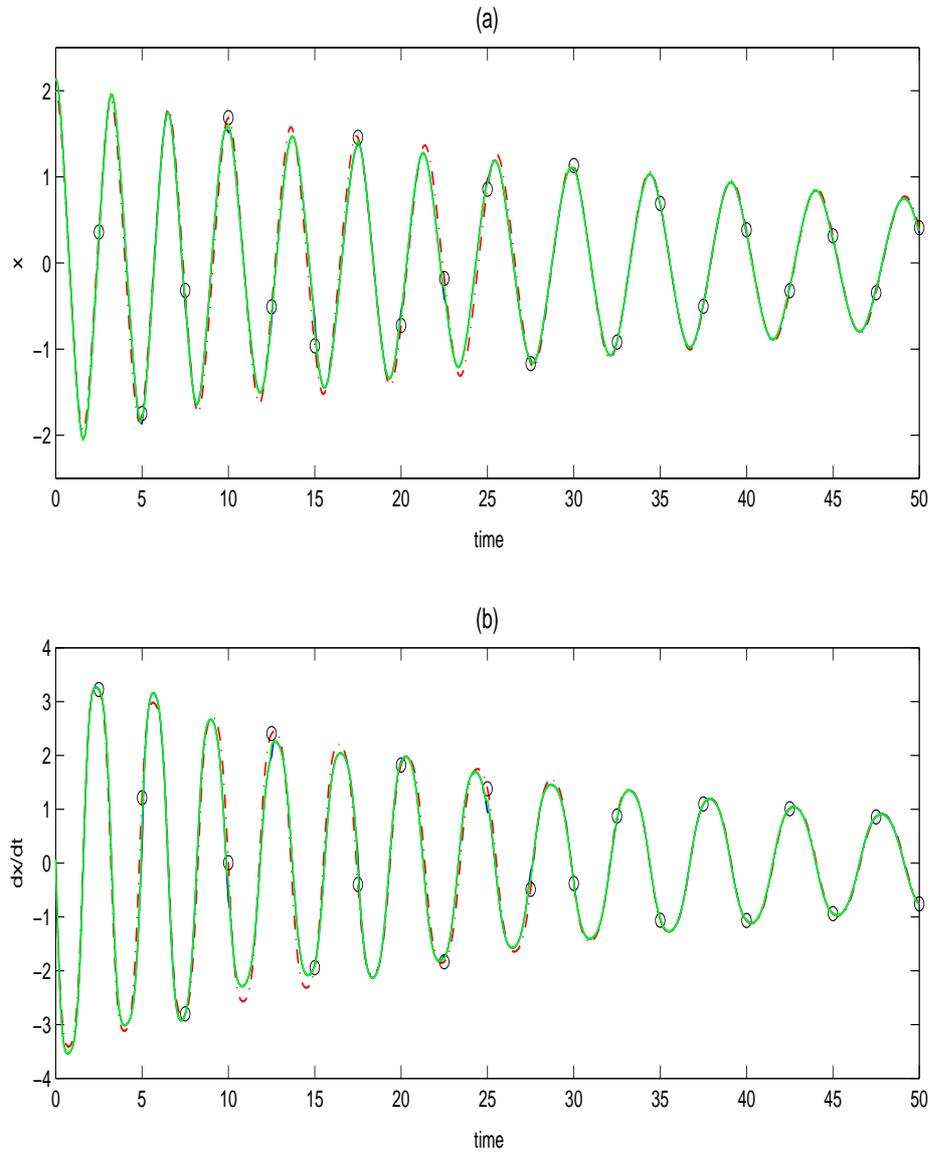


Figure 7.12: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ with initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ and perfect observations assimilated every $25\Delta t$. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

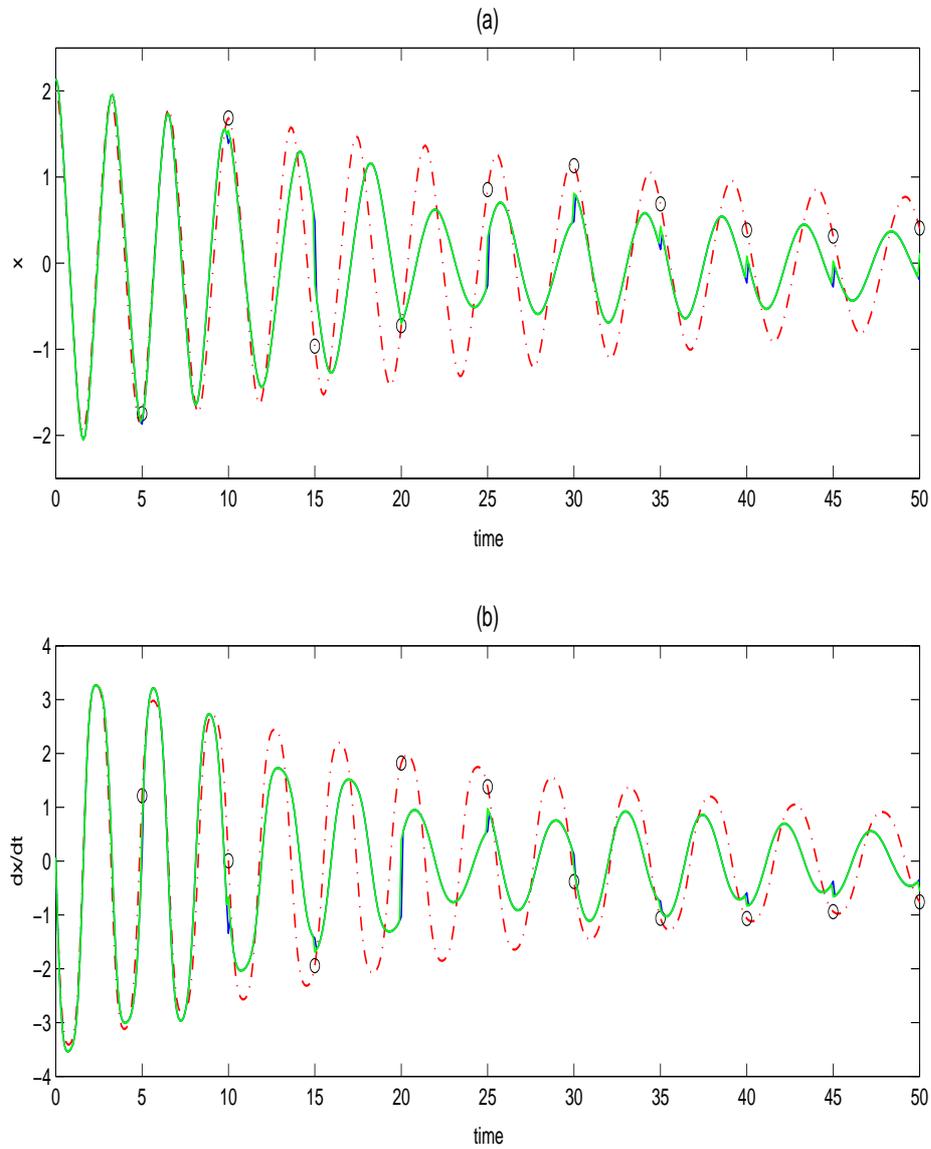


Figure 7.13: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ with initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ and perfect observations assimilated every $50\Delta t$. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

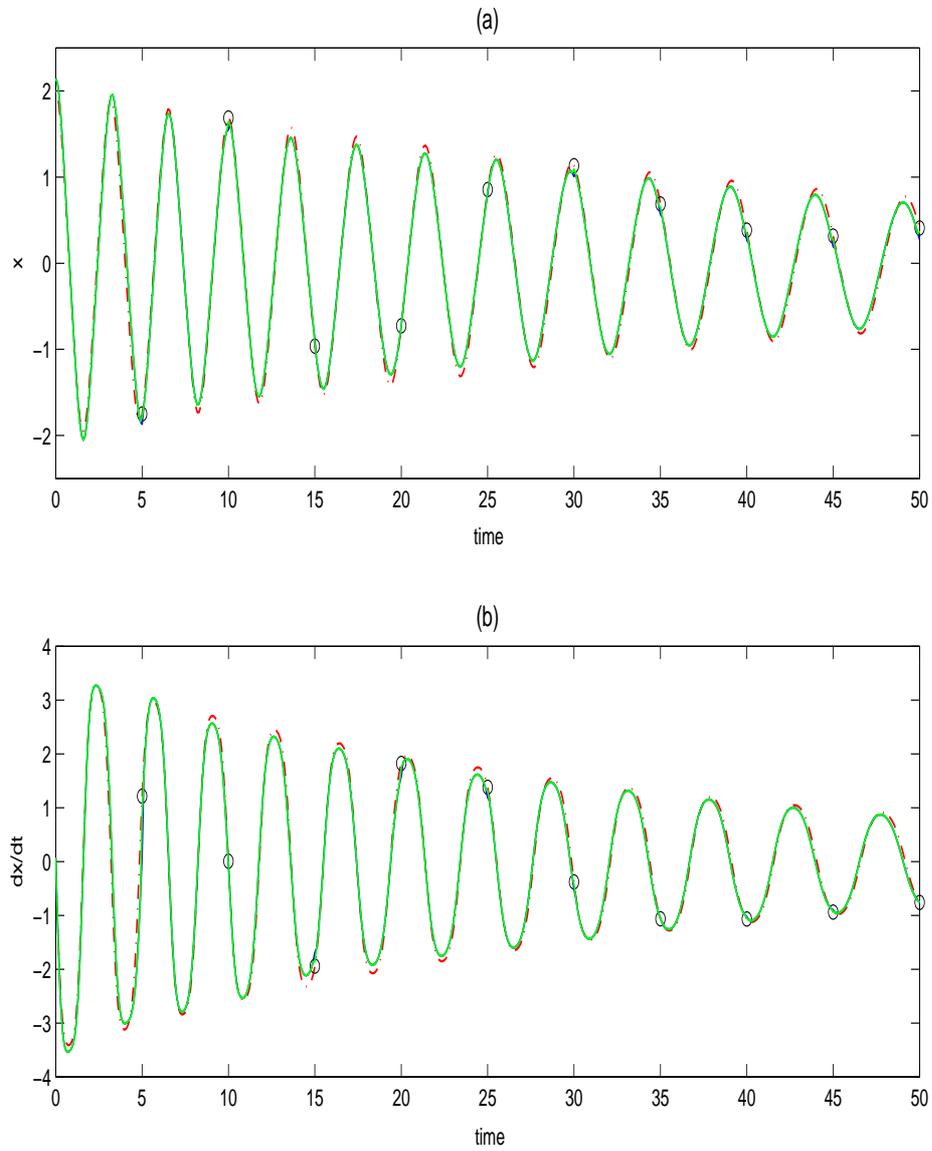


Figure 7.14: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ with initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ and perfect observations assimilated every $50\Delta t$ with inflated σ_b^2 . The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

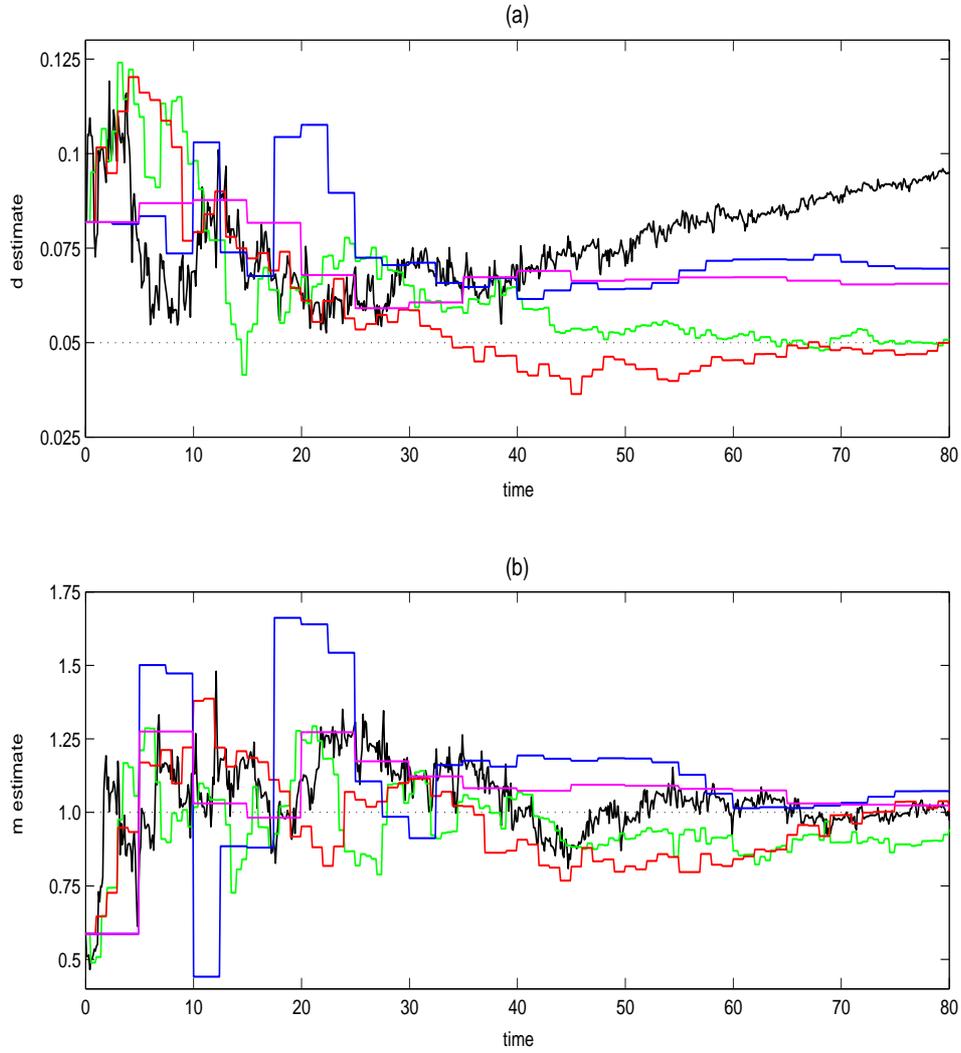


Figure 7.15: Nonlinear oscillator: imperfect observations with observation error variance $\sigma_o^2 = 0.01$ **(a)** Parameter d updates for initial estimate $d_0 = 0.081877$, **(b)** Parameter m updates for initial estimate $m_0 = 0.58617$: solid black line - observations every Δt ; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid pink line - observations every $50\Delta t$. The dotted black line indicates the true parameter value.

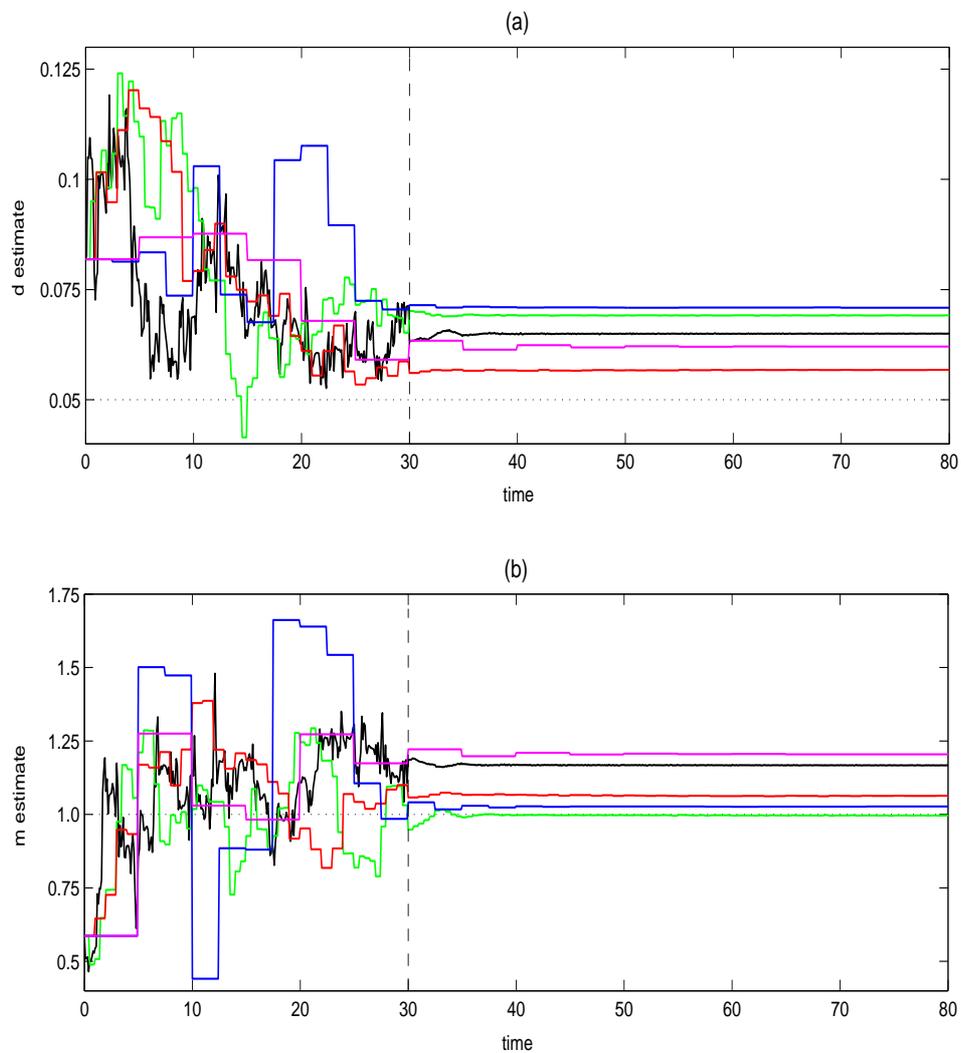


Figure 7.16: Nonlinear oscillator: imperfect observations with observation error variance $\sigma_o^2 = 0.01$ **(a)** Averaged parameter d updates for initial estimate $d_0 = 0.081877$, **(b)** Averaged parameter m updates for initial estimate $m_0 = 0.58617$: solid black line - observations every Δt ; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid pink line - observations every $50\Delta t$. The dotted black line indicates the true parameter value.

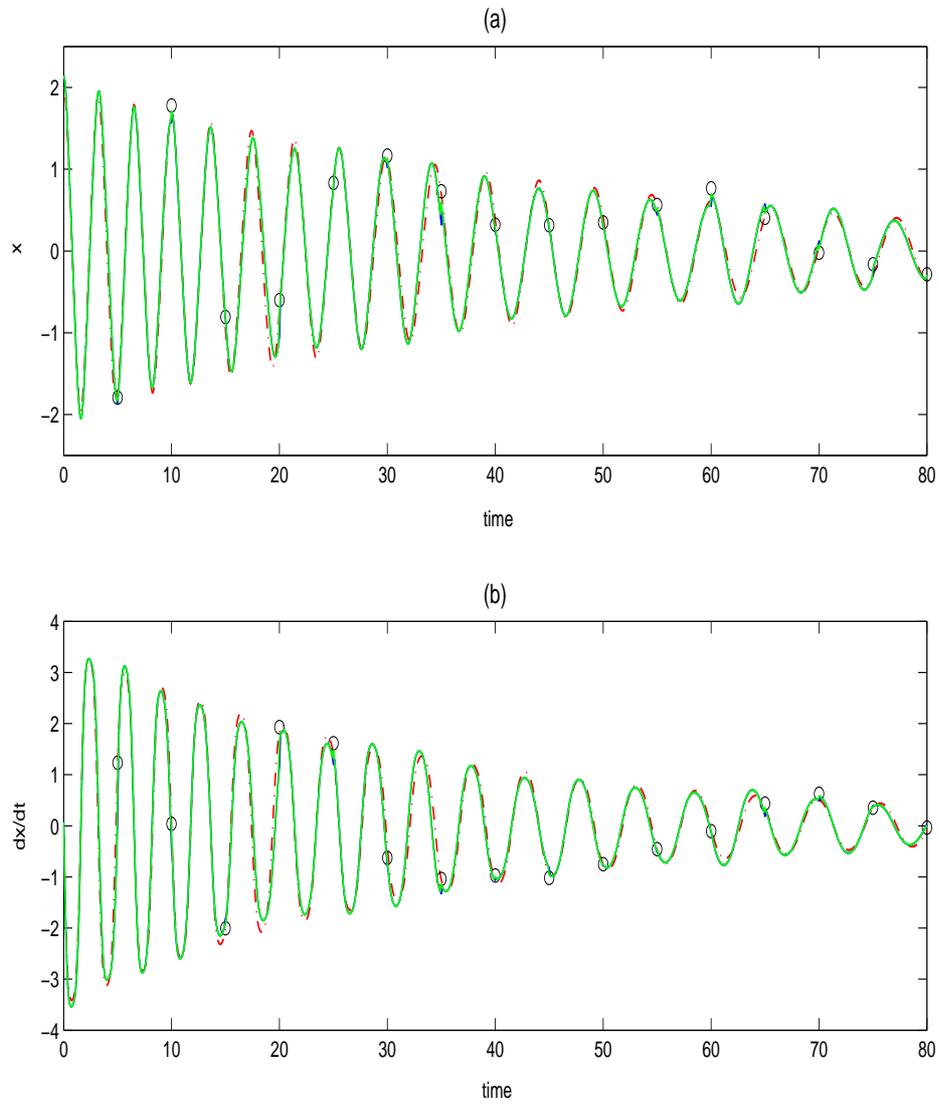


Figure 7.17: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ using initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ with noisy observations assimilated every $50\Delta t$ and $\sigma_o^2 = 0.01$, $\sigma_b^2 = 0.02$. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

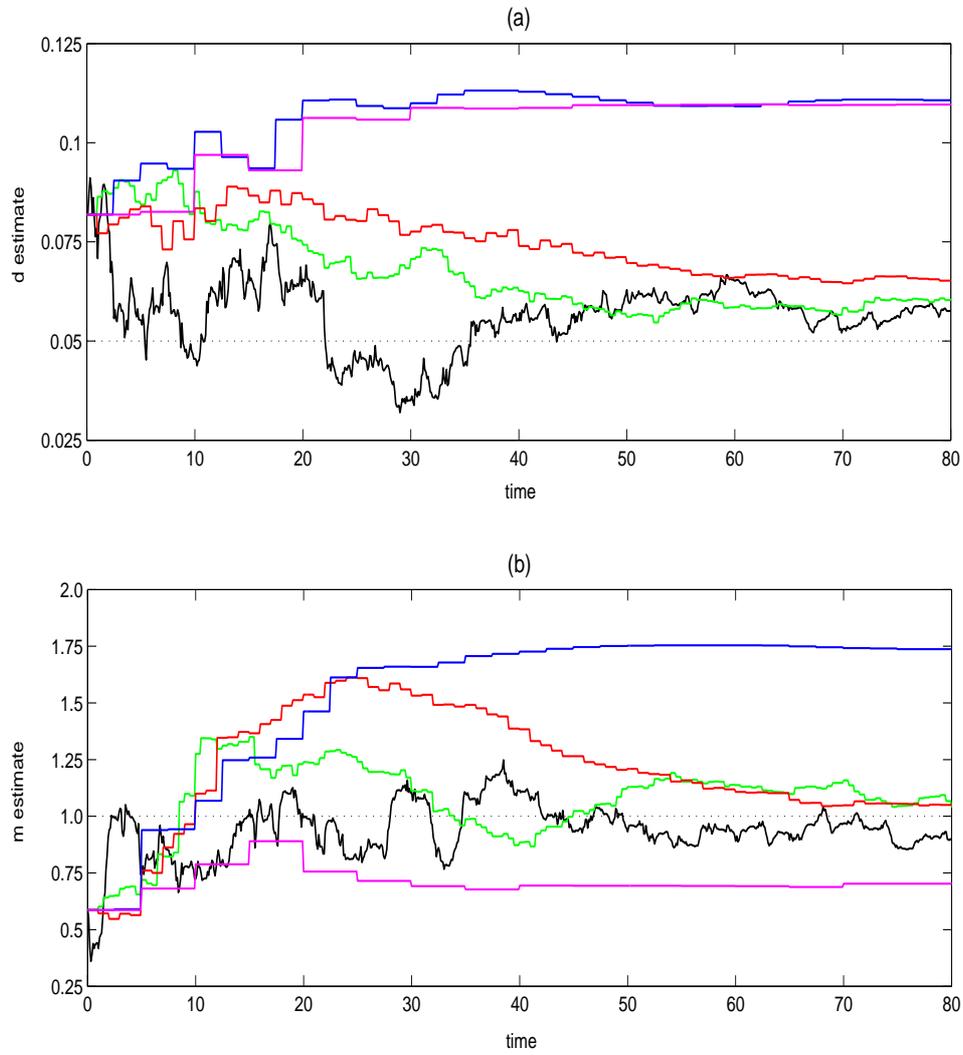


Figure 7.18: Nonlinear oscillator: imperfect observations with observation error variance $\sigma_o^2 = 0.1$ **(a)** Parameter d updates for initial estimate $d_0 = 0.081877$, **(b)** Parameter m updates for initial estimate $m_0 = 0.58617$: solid black line - observations every Δt ; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid pink line - observations every $50\Delta t$. The dotted black line indicates the true parameter value.

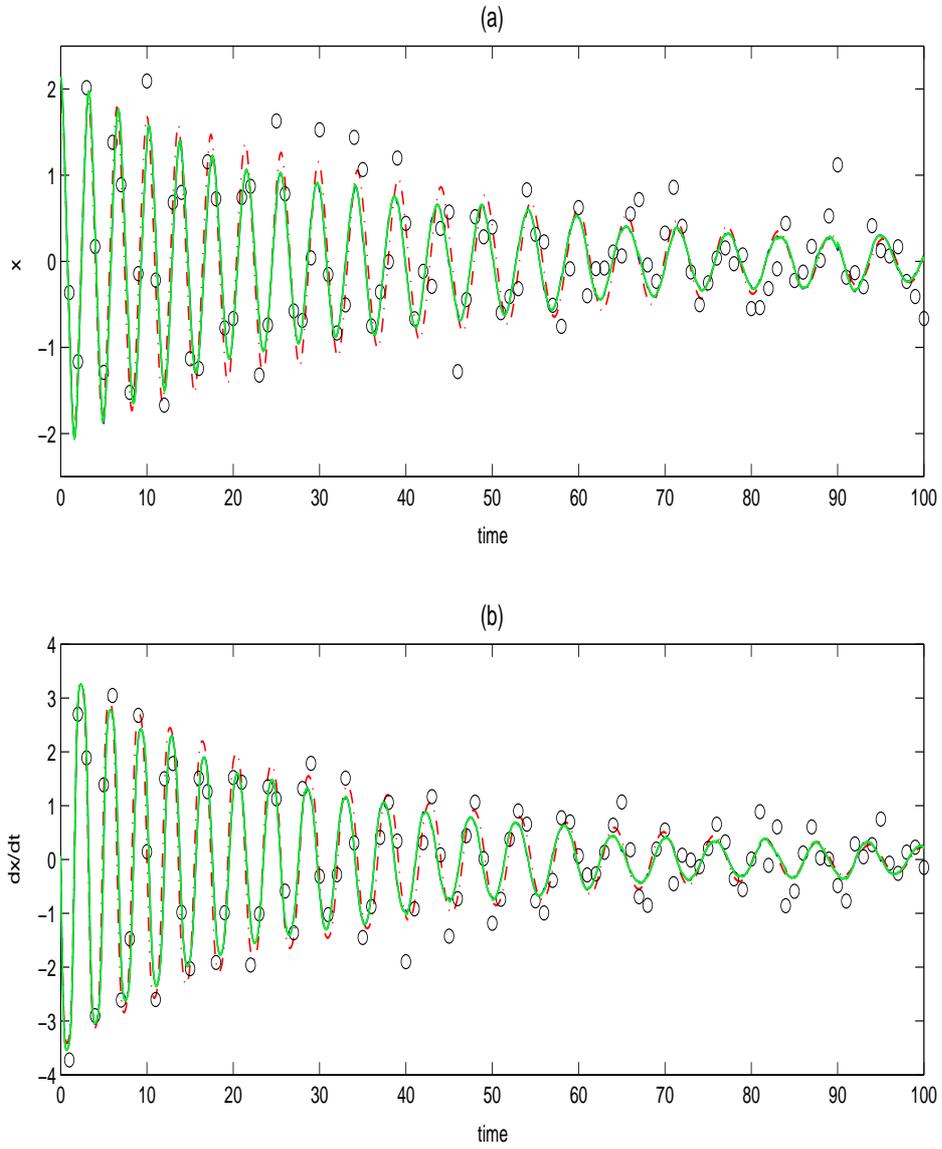


Figure 7.19: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ using initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ with noisy observations assimilated every $10\Delta t$ and $\sigma_o^2 = 0.1$. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

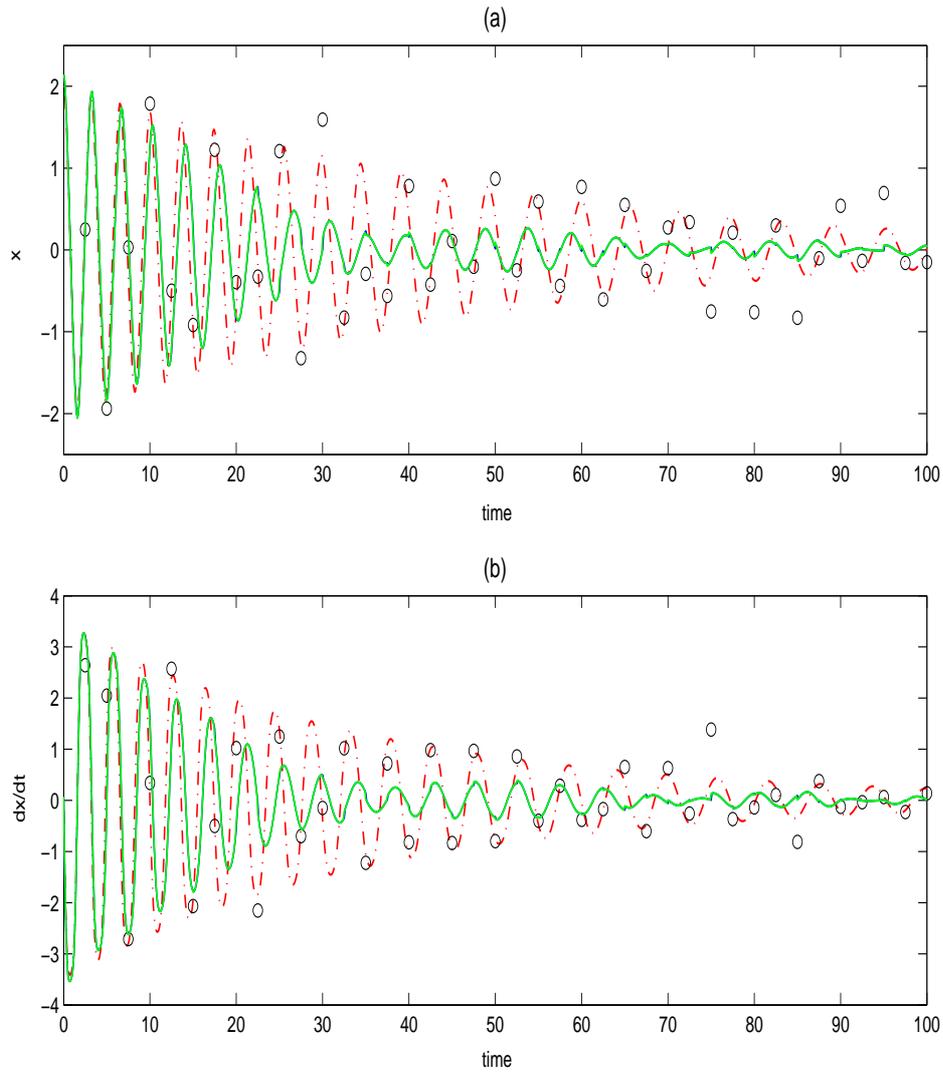


Figure 7.20: Nonlinear oscillator: analysis for x and $y = \frac{dx}{dt}$ using initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ with noisy observations assimilated every $25\Delta t$ and $\sigma_o^2 = 0.1$. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

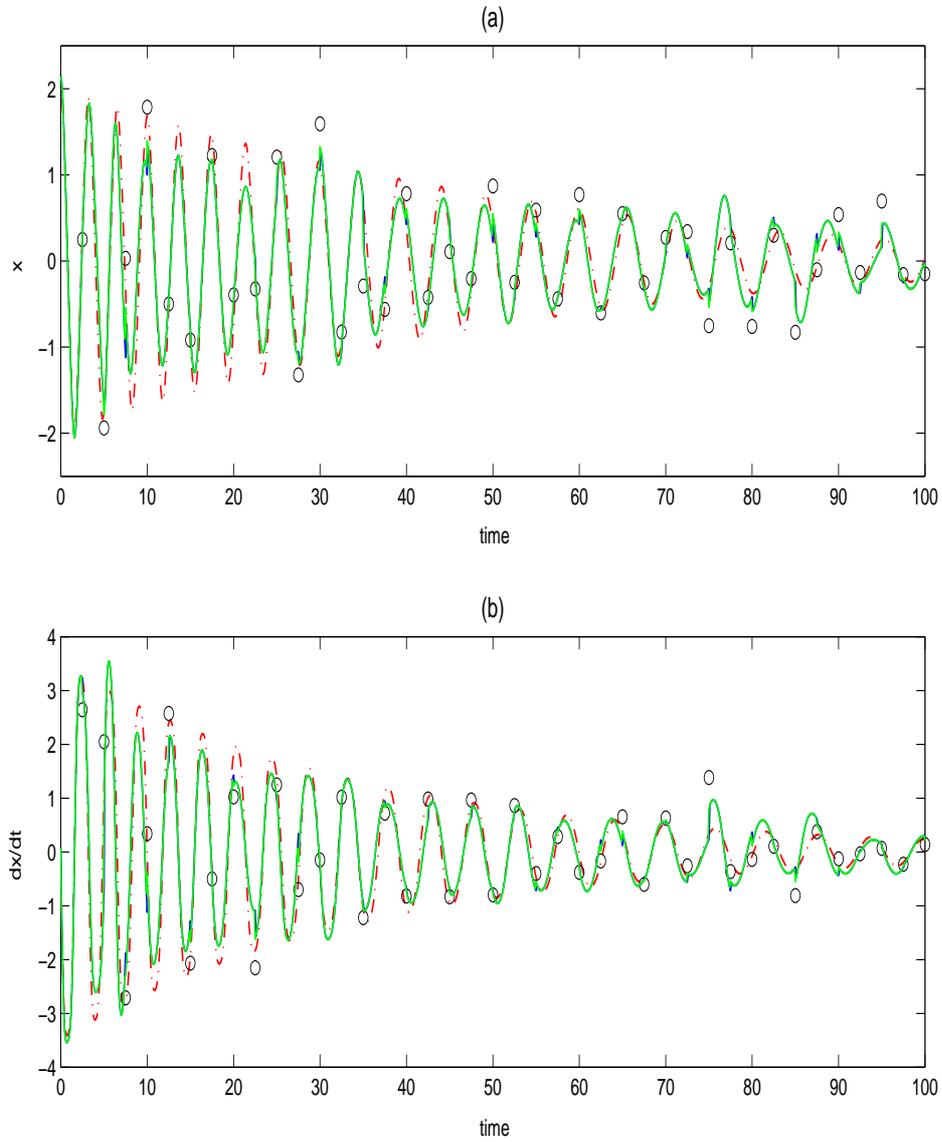


Figure 7.21: Nonlinear oscillator - overweighting the observations: analysis for x and $y = \frac{dx}{dt}$ using initial parameter estimates (i) $d_0 = 0.081877$, $m_0 = 0.58617$ with noisy observations assimilated every $25\Delta t$. The observation noise is generated using $\sigma_o^2 = 0.1$ but an observation error variance of $\sigma_o^2 = 0.01$ is used in the assimilation. The red dashed line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

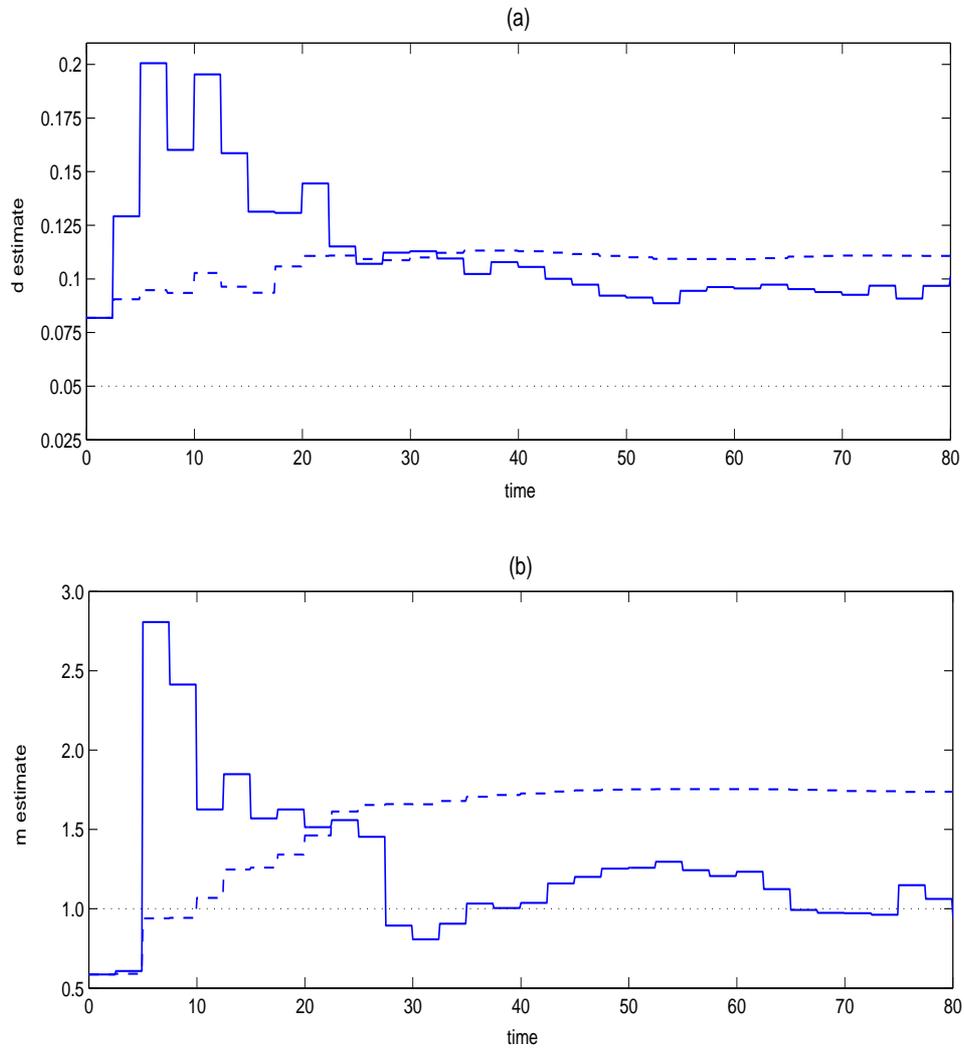


Figure 7.22: Nonlinear oscillator - overweighting the observations: imperfect observations with error variance $\sigma_o^2 = 0.1$ assimilated every $25\Delta t$. **(a)** parameter d updates for initial estimate $d_0 = 0.081877$, **(b)** parameter m updates for initial estimate $m_0 = 0.58617$. The dotted black line indicates the true parameter value. The solid blue line represents the updates obtained when an under estimated observation error variance $\sigma_o^2 = 0.01$ is used in the assimilation algorithm; the dashed blue line shows the results produced when the true observation error variance $\sigma_o^2 = 0.1$ is used.

estimates when observations are assimilated at every time step (solid black line in figure 7.15(a)). The estimates initially appear to be moving towards the correct value but at around 40 time steps they begin to increase away. The experiment was repeated with different noise simulations and different starting values for d but similar behaviour was found in every case. It is possible that the interval between assimilation time steps is insufficient for the model to adjust to the updated value of d . A further hypothesis is that this behaviour is related to the role of d in the model equations. The parameter d determines how quickly the solution becomes damped. As we move forward in time the amplitude of the solution decreases, the relative size of the observational noise therefore increases causing greater misrepresentation of the true amplitude and making it harder to identify the true d value.

We found that this behaviour could be remedied by averaging the estimates as is illustrated in figures 7.16 (a) and (b). For observation intervals of Δt to $25\Delta t$ the parameter estimates were averaged over a moving time window of 50 timesteps starting at $t = 30$. For observations every $50\Delta t$ this was increased to 100 time steps. The averaging produces more stable estimates for the parameters which in turn gives greater stability to the forecast model.

In this example, the value prescribed for σ_o^2 is relatively small and so the observational noise has very little impact on the overall quality of the state analysis as can be seen by comparing figure 7.17 with figure 7.14 (note that figure 7.17 shows the state analysis obtained with the unaveraged parameter estimates). We see a much greater effect on the state analysis if the observation error variance is increased to $\sigma_o^2 = 0.1$. As figures 7.18 (a) and (b) illustrate, in terms of the parameters, for observation intervals Δt , $5\Delta t$ and $10\Delta t$, we are still able to estimate both the parameters to a reasonable level of accuracy; although the fluctuations in the estimates are larger. Note the greater range in scale in figures 7.18 (a) and (b) compared to figure 7.15.

When an observation interval of $10\Delta t$ is used, there are initially large errors in the estimated values of parameter m . However, this is gradually corrected as more observations are processed and the m estimates eventually converge towards the true value. The d estimates also take slightly longer to reach the true d . This behaviour effects the analysis for x and y as shown in figure 7.19. Initially, amplitude and phase errors

begin to develop in the predicted solution, however once the d and m estimates have stabilised (at around $t = 70$) the analysis for x and y recovers and becomes a much better approximation of the truth.

There is a big jump in error when the observation frequency is decreased to $25\Delta t$ and $50\Delta t$. The high d estimates mean that the model very quickly becomes over-damped. The increased time between assimilations mean that, unlike the $10\Delta t$ case, the scheme is unable to recover and as a result the state analysis becomes extremely poor. By increasing the observation errors we now have $\sigma_b^2 \ll \sigma_o^2$. The analysis in figure 7.20 shows that insufficient weight is being given to the observations. Taking a similar approach to the $50\Delta t$ perfect observation case in which the background error variance was inflated, we found that the state estimates could be improved by increasing the weight given to the observations. Figure 7.21 shows the x and y analysis for observation frequencies $25\Delta t$ when variance $\sigma_o^2 = 0.1$ is used to simulate the observation noise but a observation error variance of $\sigma_o^2 = 0.01$ is used in the assimilation. The parameter updates are shown in figures 7.22 (a) and (b) along with previous estimates obtained using error variance $\sigma_o^2 = 0.1$. The errors in the damping parameter d are slightly reduced but the estimates are still too high. There is a substantial improvement in the parameter m estimates which now converge close to the true value. The same approach was tried with an observation frequency of $50\Delta t$. This failed to produce any real improvement in the parameter estimates but did produce a notable improvement in the state analysis.

7.3 The Lorenz 63 equations

The Lorenz equations is the name given to a system of first order differential equations describing a simple nonlinear dynamical system that exhibits chaotic behaviour. The system was originally derived from a model of fluid convection and consist of the three coupled, nonlinear ordinary differential equations [60]

$$\dot{x} = -\sigma(x - y), \quad (7.35)$$

$$\dot{y} = \rho x - y - xz, \quad (7.36)$$

$$\dot{z} = xy - \beta z, \quad (7.37)$$

where $x = x(t)$, $y = y(t)$ and $z = z(t)$ and σ , ρ and β are real, positive parameters.

The strong nonlinearity of these equations means that the model solution is extremely sensitive to perturbations in the initial conditions and parameters. For this reason, the model is often used as a framework for examining the properties of data assimilation methods when applied to highly nonlinear dynamical systems [21], [69].

The origin is a stationary point for all parameter values. When $\rho > 1$ there are two other stationary points

$$\left(\pm\sqrt{\beta(\rho-1)}, \pm\sqrt{\beta(\rho-1)}, \rho-1 \right).$$

For these experiments we set the ‘true’ parameters at $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$. These are the classic values first used by Lorenz. At these values all three equilibrium points are unstable and give rise to chaotic solutions [95].

To investigate the applicability of our new hybrid assimilation scheme to this system we adapt a pre-existing Matlab routine written by M.J Martin [64], [62]. The code was originally developed as a training aid to illustrate the application of sequential data assimilation schemes to state estimation in simplified models. A copy of the original, unmodified code can be obtained from [61]. The program solves equations (7.35)-(7.37) numerically using the same second order Runge-Kutta method as was used in section 7.2. The discrete system is given by

$$x_{k+1} = x_k + \sigma \frac{\Delta t}{2} \left[2(y_k - x_k) + \Delta t(\rho x_k - y_k - x_k z_k) - \sigma \Delta t(y_k - x_k) \right], \quad (7.38)$$

$$\begin{aligned} y_{k+1} = & y_k + \frac{\Delta t}{2} \left[\rho x_k - y_k - x_k z_k + \rho(x_k + \sigma \Delta t(y_k - x_k)) - y_k - \Delta t(\rho x_k - y_k - x_k z_k) \right. \\ & \left. - (x_k + \sigma \Delta t(y_k - x_k))(z_k + \Delta t(x_k y_k - \beta z_k)) \right] \end{aligned} \quad (7.39)$$

$$\begin{aligned} z_{k+1} = & z_k + \frac{\Delta t}{2} \left[x_k y_k - \beta z_k + (x_k + \Delta t \sigma(y_k - x_k))(y_k + \Delta t(\rho x_k - y_k - x_k z_k)) \right. \\ & \left. - \beta(z_k + \Delta t(x_k y_k - \beta z_k)) \right]. \end{aligned} \quad (7.40)$$

We define the state and parameter vectors

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix}, \quad \text{and} \quad \mathbf{p}_k = \begin{pmatrix} \sigma_k \\ \rho_k \\ \beta_k \end{pmatrix}. \quad (7.41)$$

giving the augmented system model

$$\begin{aligned}\mathbf{w}_{k+1} &= \tilde{\mathbf{f}}(\mathbf{w}_k) \\ &= \begin{pmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix}.\end{aligned}\quad (7.42)$$

where $\mathbf{w} \in \mathbb{R}^6$ is the augmented state vector, $\mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is the state evolution model given by (7.38)-(7.40) and $\mathbf{p}_{k+1} = \mathbf{p}_k$ is the parameter evolution model.

7.3.1 State-parameter cross covariance

For the Lorenz model, the Jacobian matrix \mathbf{N}_k is given by

$$\begin{aligned}\mathbf{N}_k &= \left(\begin{array}{ccc} \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial \sigma} & \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial \rho} & \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial \beta} \end{array} \right) \Big|_{\mathbf{x}_k^a, \mathbf{p}_k^a} \\ &= \left(\begin{array}{ccc} \frac{\partial x_{k+1}}{\partial \sigma} & \frac{\partial x_{k+1}}{\partial \rho} & \frac{\partial x_{k+1}}{\partial \beta} \\ \frac{\partial y_{k+1}}{\partial \sigma} & \frac{\partial y_{k+1}}{\partial \rho} & \frac{\partial y_{k+1}}{\partial \beta} \\ \frac{\partial z_{k+1}}{\partial \sigma} & \frac{\partial z_{k+1}}{\partial \rho} & \frac{\partial z_{k+1}}{\partial \beta} \end{array} \right) \Big|_{\sigma_k, \rho_k, \beta_k},\end{aligned}\quad (7.43)$$

and can be computed by differentiating the discrete equations (7.38), (7.39) and (7.40) with respect to each of the parameters as described in appendix D.

We assume that σ , ρ and β are uncorrelated and set

$$\mathbf{B}_{\mathbf{pp}} = \begin{pmatrix} \sigma_\sigma^2 & 0 & 0 \\ 0 & \sigma_\rho^2 & 0 \\ 0 & 0 & \sigma_\beta^2 \end{pmatrix},\quad (7.44)$$

where σ_σ^2 , σ_ρ^2 and σ_β^2 are the error variances of the parameters σ , ρ and β respectively.

The state-parameter cross covariance matrix at t_{k+1} is then given by

$$\begin{aligned}\mathbf{B}_{\mathbf{xp}_{k+1}} &= \mathbf{N}_k \mathbf{B}_{\mathbf{pp}} \\ &= \left(\begin{array}{ccc} \sigma_\sigma^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \sigma} & \sigma_\rho^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \rho} & \sigma_\beta^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \beta} \end{array} \right) \Big|_{\mathbf{x}_k^a, \mathbf{p}_k^a}.\end{aligned}\quad (7.45)$$

7.3.2 Assimilation experiments

Again we test our scheme by running identical twin experiments. The ‘true’ solution is taken to be that given by the discrete equations (7.38) - (7.40) with model time step $\Delta t = 0.01$ and initial conditions $x_0 = -5.4458$, $y_0 = -5.4841$ and $z_0 = 22.5606$. The solutions for x and z are illustrated in figure 7.23 for $t \in [0, 30]$.

Initial estimates of the three parameters $\tilde{\sigma}$, $\tilde{\rho}$ and $\tilde{\beta}$ are generated by adding random noise with variance equal to 20% of the true value to each of σ , ρ and β . The components of the initial model background state vector \mathbf{x}_0^b are taken equal to the true initial values plus random noise. This noise is taken from a Gaussian distribution with zero mean and variance $\sigma_b^2 = 0.1$. The state background error covariance matrix is taken as

$$\mathbf{B}_{\mathbf{xx}} = \sigma_b^2 \mathbf{I}, \quad \mathbf{I} \in \mathbb{R}^{3 \times 3}, \quad (7.46)$$

with inflated error variance $\sigma_b^2 = 1.0$.

Observations of x , y and z are taken from the true solution and assimilated sequentially at regular time intervals as described in algorithm 6.1. The augmented observation operator $\tilde{\mathbf{H}}$ and observation error covariance matrix \mathbf{R} for this system are given by (7.33) and (7.34) with $\mathbf{I} \in \mathbb{R}^{3 \times 3}$.

Although by adding the parameters to the state vector we double the dimension of the system, its size is still relatively small. We therefore use equation (3.12) to compute the analysis directly.

7.3.3 Results

Perfect observations

Once again we find that the hybrid scheme performs extremely well. Figure 7.24 shows the parameter updates for model runs with observations at decreasing temporal frequency. For this example, the initial parameter values are $\tilde{\sigma} = 11.0311$, $\tilde{\rho} = 30.1316$ and $\tilde{\beta} = 1.6986$ and the observation error variance is set at $\sigma_o^2 = 0.01$. For observation frequencies $5\Delta t$, $10\Delta t$ and $20\Delta t$ the estimates of ρ and β converge to their true values

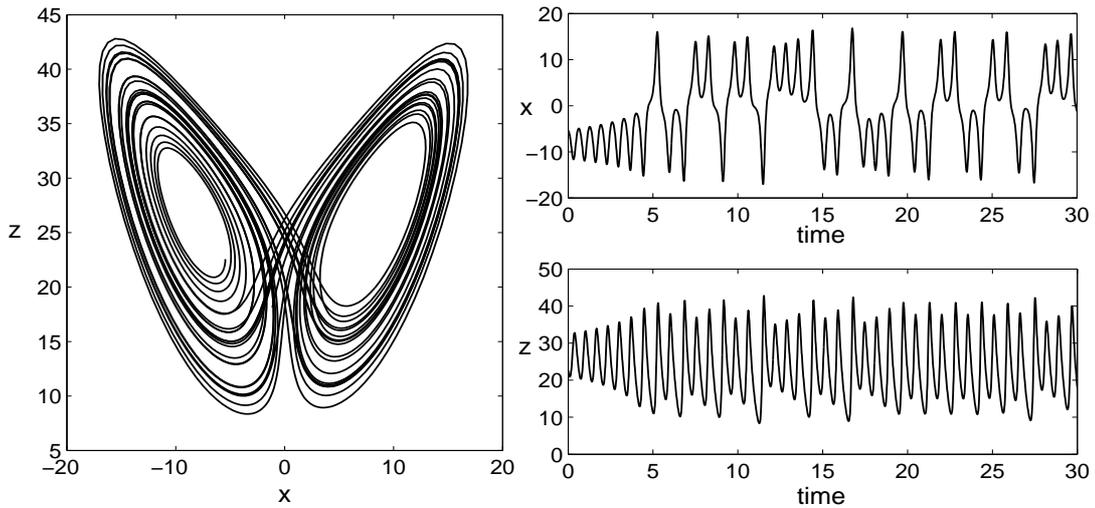


Figure 7.23: Lorenz equations: computed numerical solution for x and z .

very rapidly. The updating of σ is much slower but the correct value is eventually recovered to a high level of accuracy. There are no visible differences between the model and true solutions for x , y and z at these frequencies and so we do not show the results here. When the observation interval is increased to $30\Delta t$ the scheme takes longer to stabilise and there are much larger initial deviations in the parameter estimates. The updating of the state variables x and z is shown in figure 7.25. Although initially there are some quite large differences between the predicted and true solutions, at around $t = 10 - 15$ the model appears to have stabilised and beyond $t = 25$ the true solution is reproduced near perfectly. If the time period between assimilations is further increased to $40\Delta t$ the scheme completely fails to find the correct parameter values and the state analysis is poor across the entire assimilation window (figure 7.26).

Noisy observations

The effect of observational errors was examined by re-running the above experiments with random Gaussian noise added to the observations. We used observation error variances $\sigma_o^2 = 0.01$, $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ and assimilated the observations at increasing time intervals as in the perfect observations example. Figures 7.27 and 7.28 show parameter updates obtained when the observation error variances were set at $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ respectively. The results for $\sigma_o^2 = 0.01$ are not shown as the convergence and quality of the parameter and state estimates is very similar to

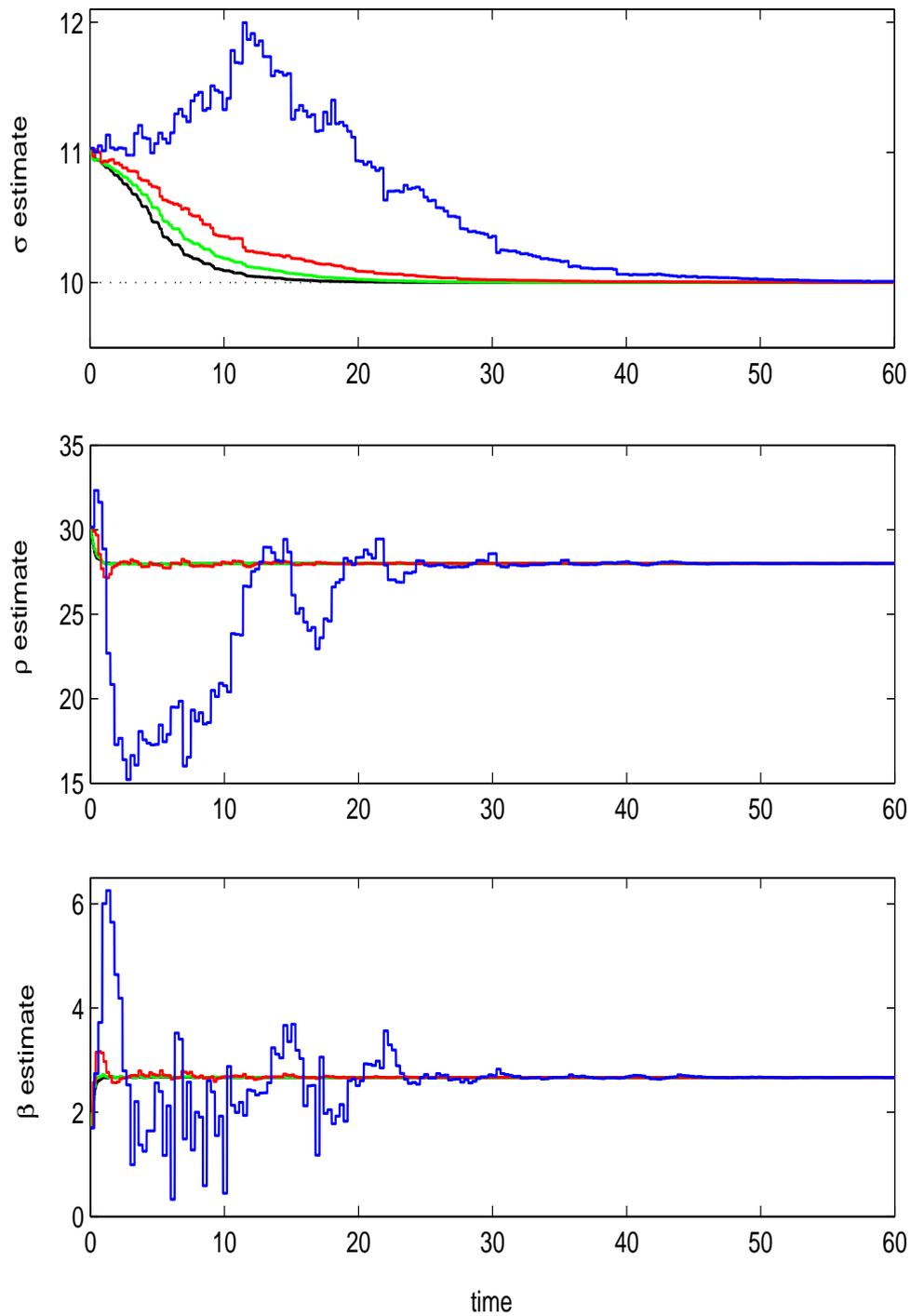


Figure 7.24: Lorenz equations: Parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$. Solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$; solid blue line - observations every $30\Delta t$. The dotted black line indicates the true parameter value.

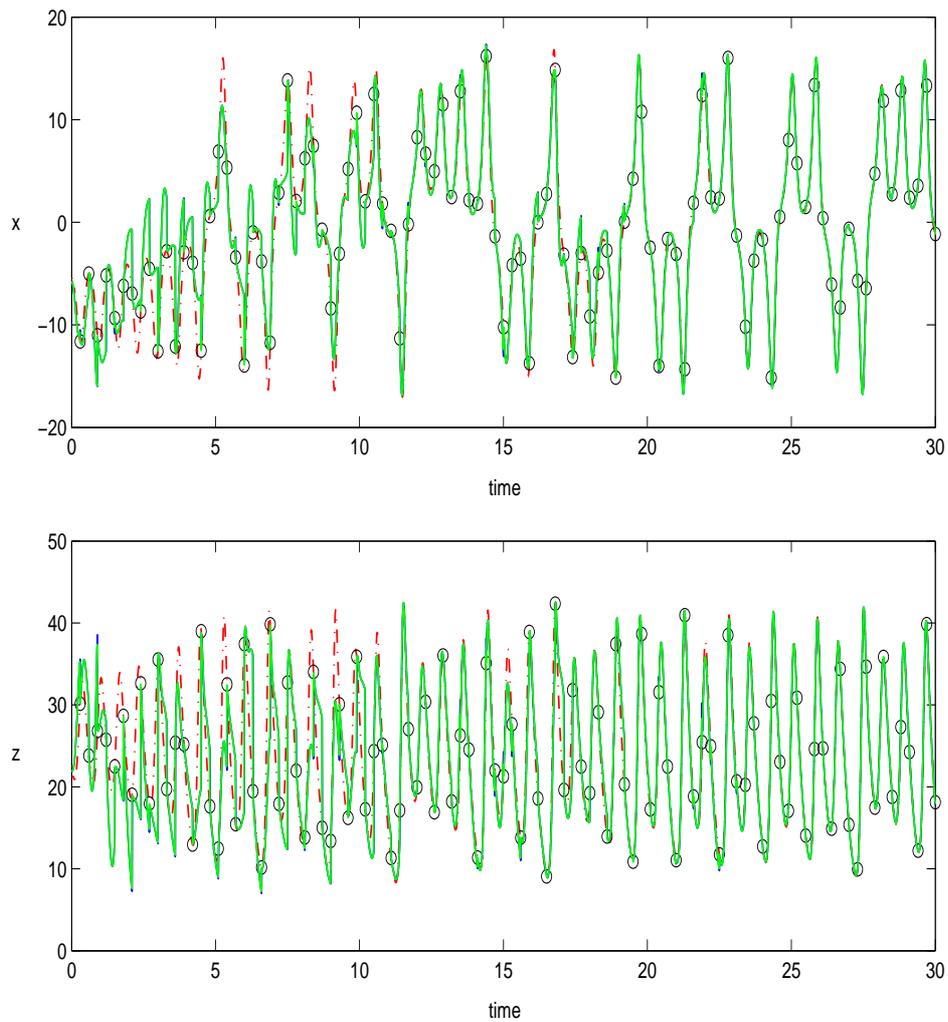


Figure 7.25: Lorenz equations: analysis for x and z using initial parameter estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$ with perfect observations assimilated every $30\Delta t$. The red dot-dash line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

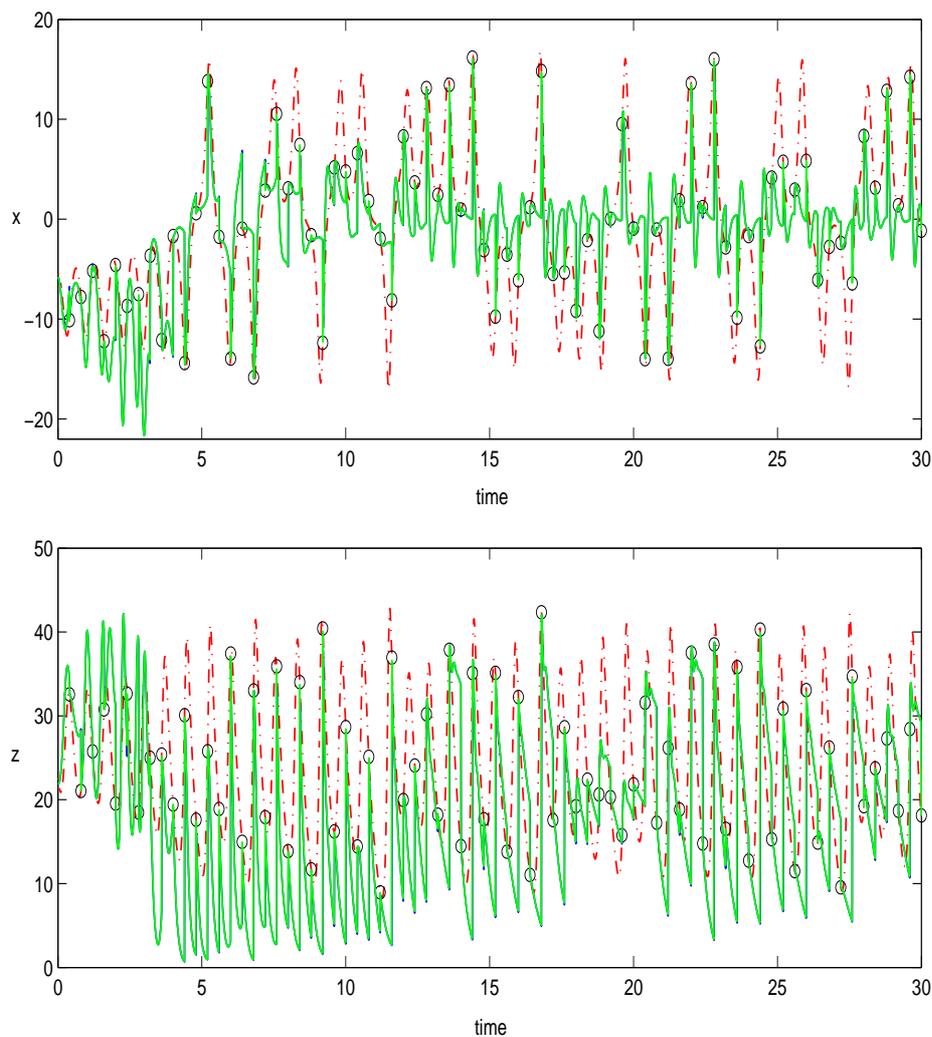


Figure 7.26: Lorenz equations: analysis for x and z using initial parameter estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$ with perfect observations assimilated every $40\Delta t$. The red dot-dash line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

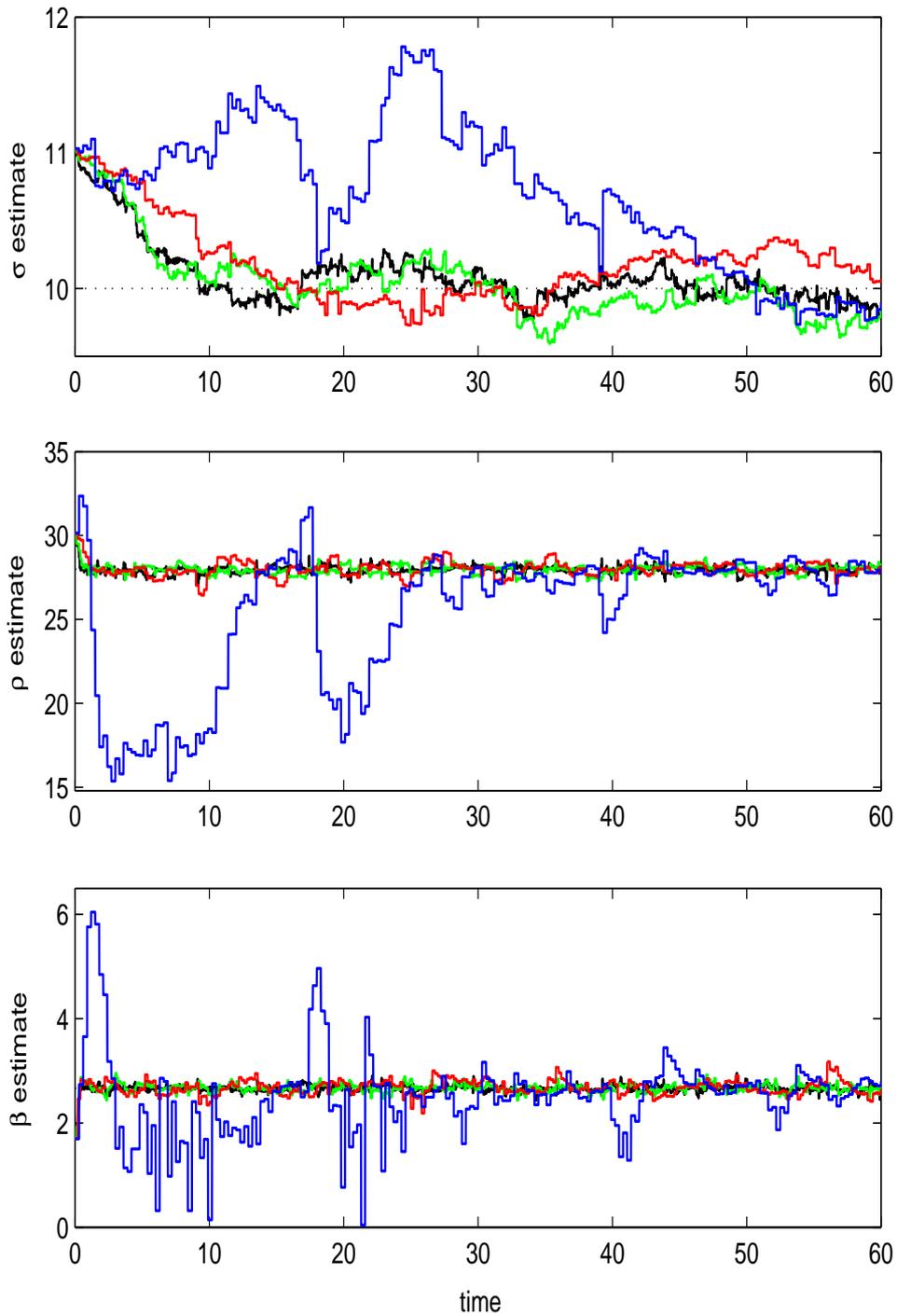


Figure 7.27: Lorenz equations: imperfect observations with observation error variance $\sigma_o^2 = 0.1$. Parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$. Solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$; solid blue line - observations every $30\Delta t$. The dotted black line indicates the true parameter value.

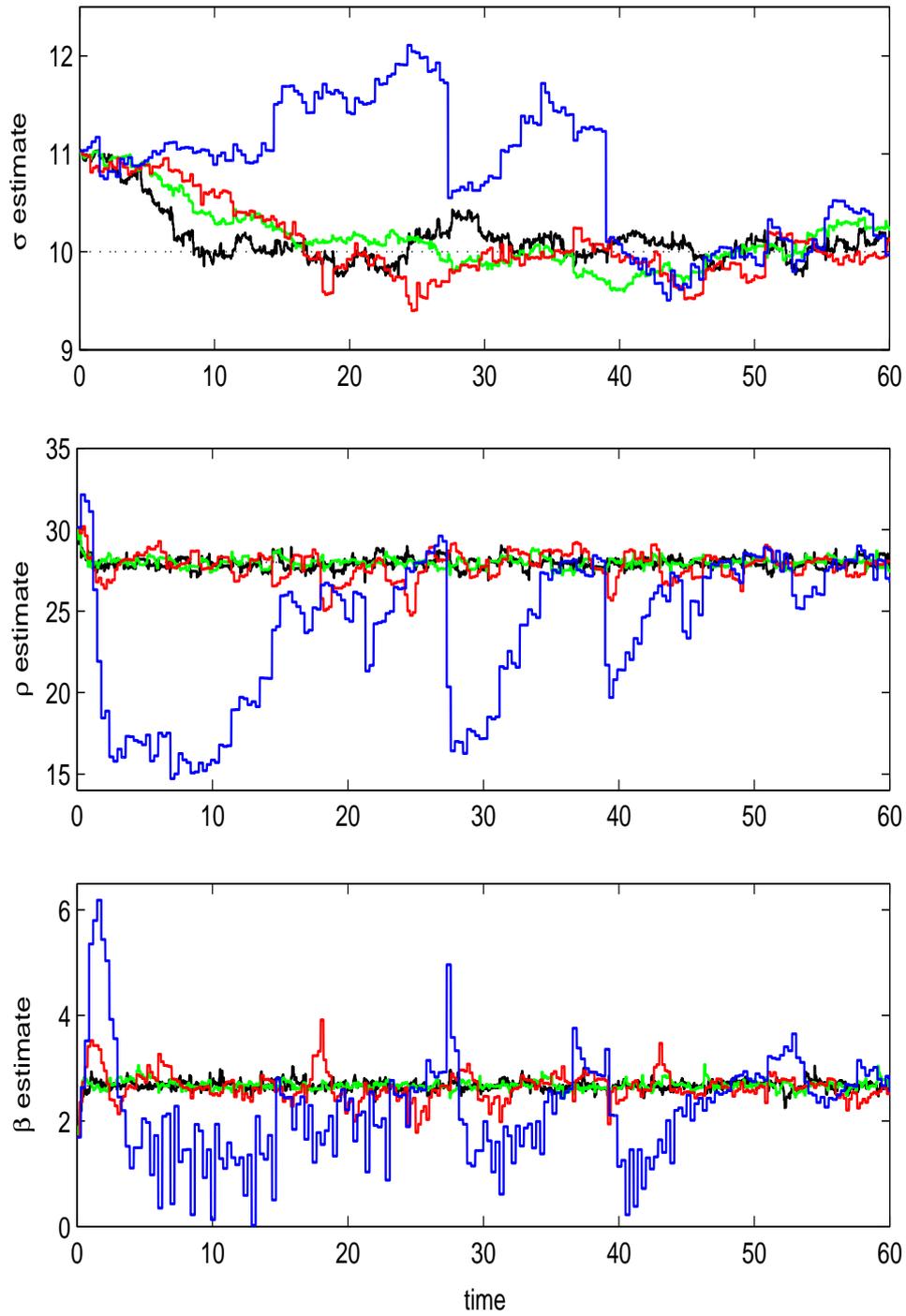


Figure 7.28: Lorenz equations: imperfect observations with observation error variance $\sigma_o^2 = 0.25$. Parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$. Solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$; solid blue line - observations every $30\Delta t$. The dotted black line indicates the true parameter value.

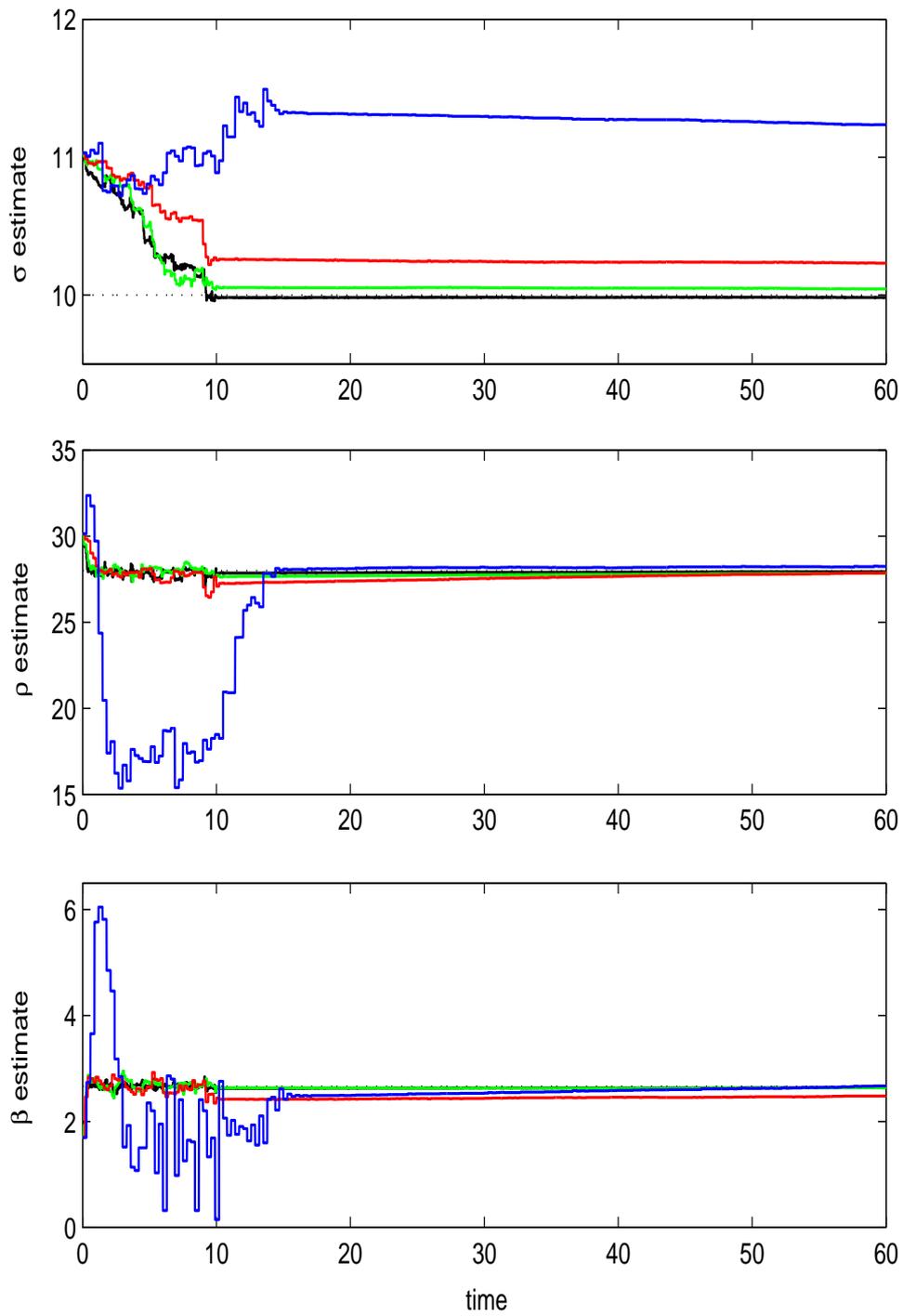


Figure 7.29: Lorenz equations: imperfect observations with observation error variance $\sigma_o^2 = 0.1$. Averaged parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$. Solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$; solid blue line - observations every $30\Delta t$. The dotted black line indicates the true parameter value.

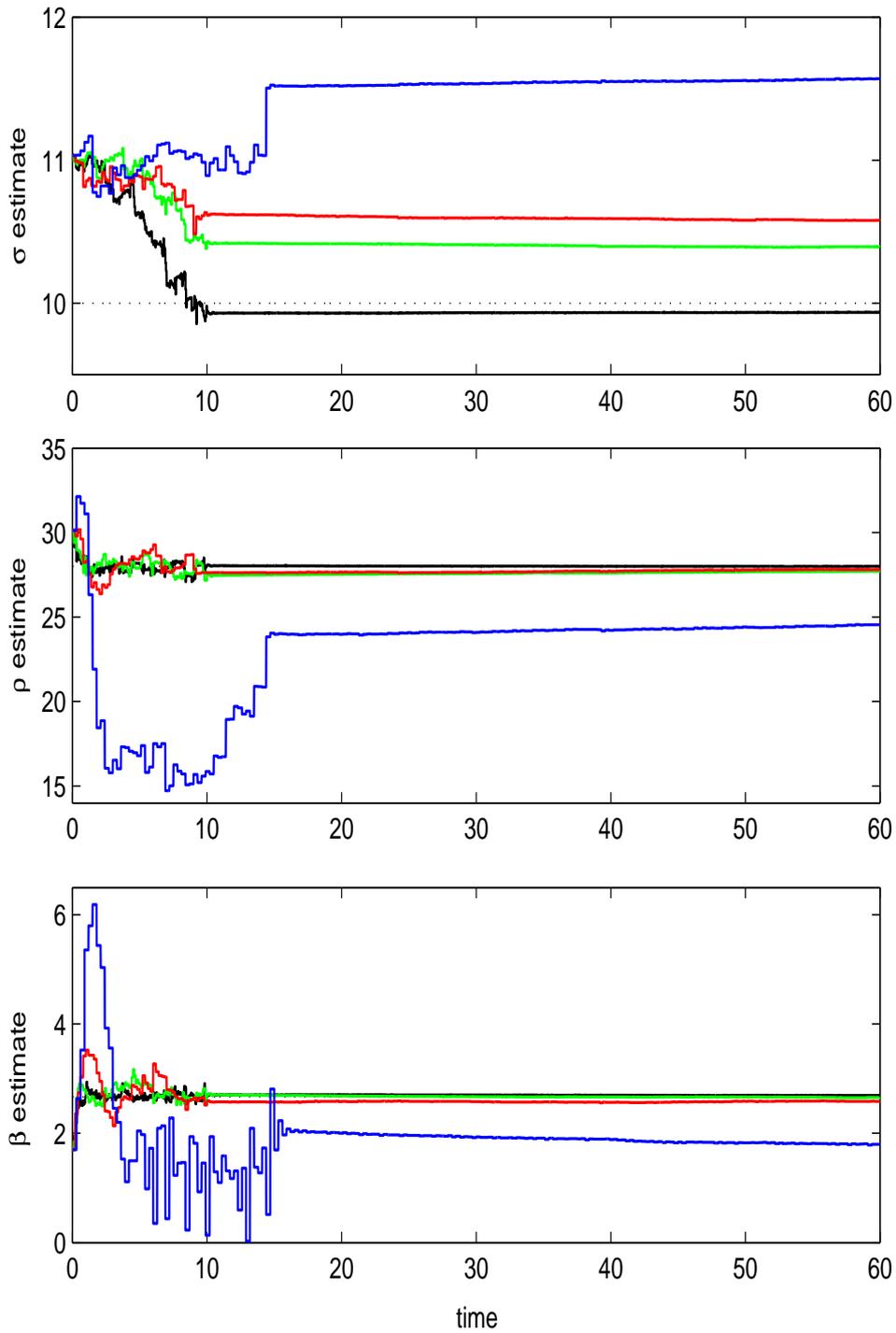


Figure 7.30: Lorenz equations: imperfect observations with observation error variance $\sigma_o^2 = 0.25$. Averaged parameter updates for initial estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$. Solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$; solid blue line - observations every $30\Delta t$. The dotted black line indicates the true parameter value.

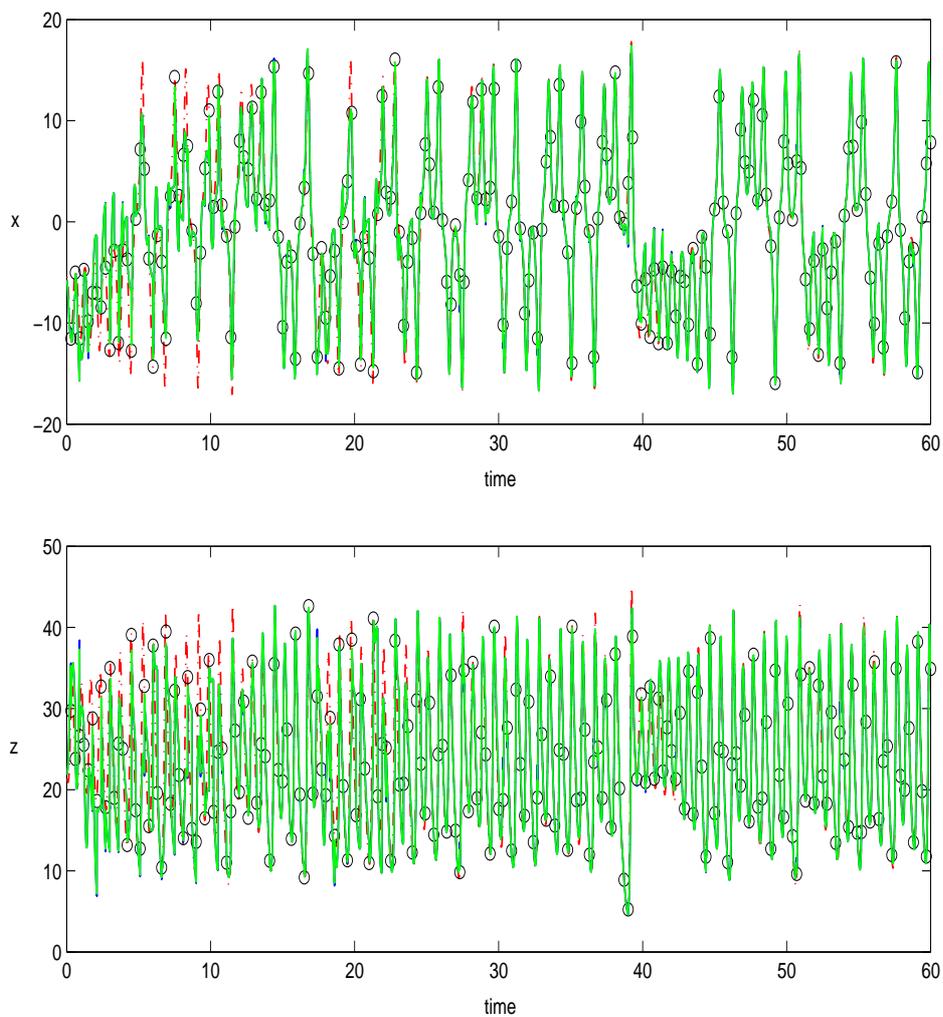


Figure 7.31: Lorenz equations: analysis for x and z using initial parameter estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$ with noisy observations assimilated every $30\Delta t$ and $\sigma_o^2 = 0.1$. The red dot-dash line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

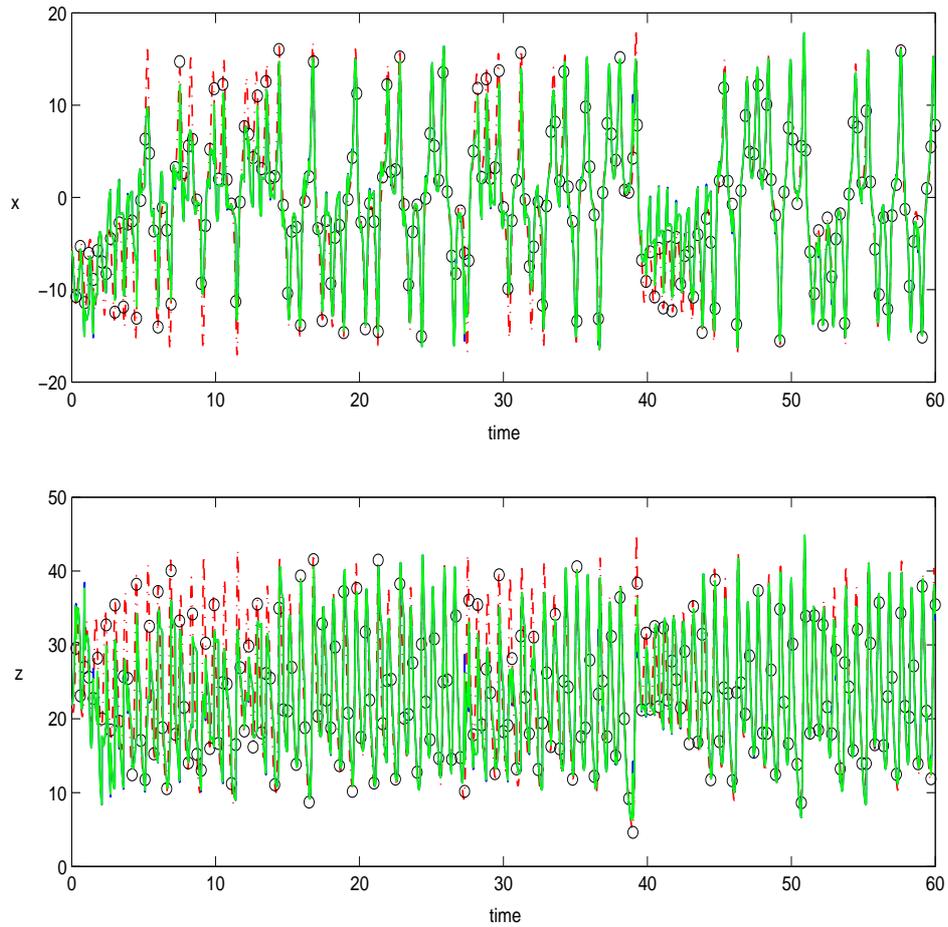


Figure 7.32: Lorenz equations: analysis for x and z using initial parameter estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$ with noisy observations assimilated every $30\Delta t$ and $\sigma_o^2 = 0.25$. The red dot-dash line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

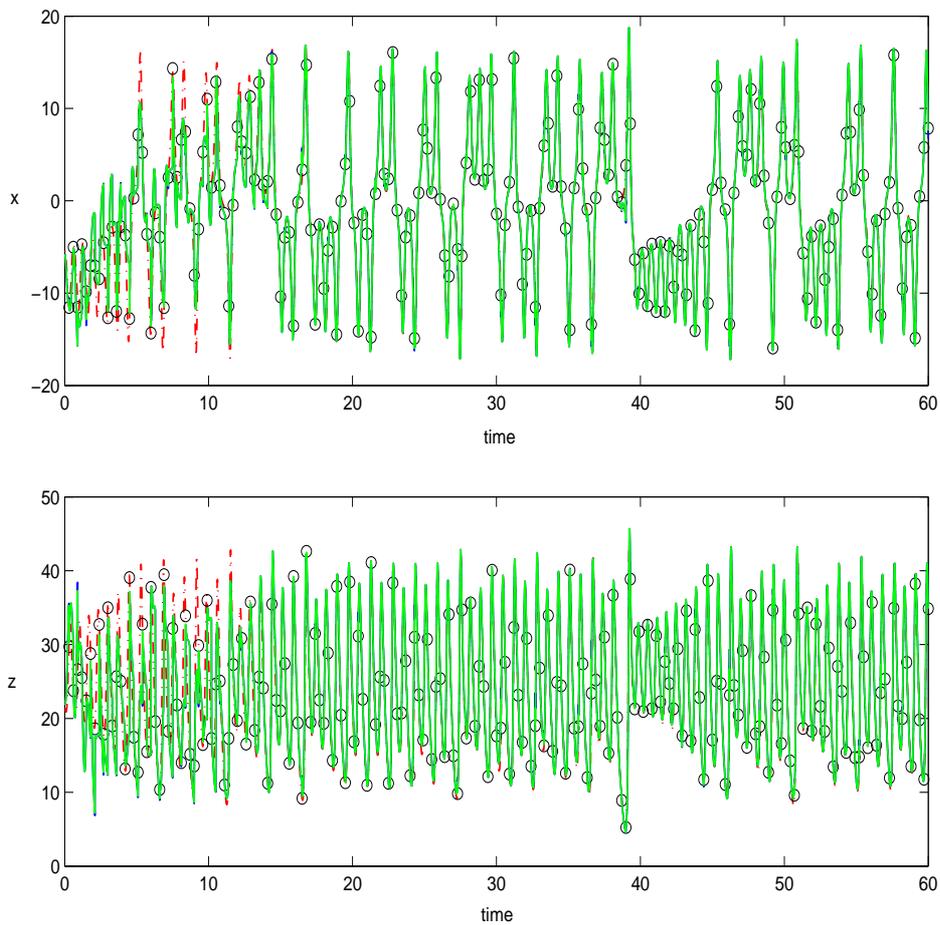


Figure 7.33: Lorenz equations - time averaged parameter estimates: analysis for x and z with initial parameter estimates $\sigma = 11.0311$, $\rho = 30.1316$ and $\beta = 1.6986$ with noisy observations assimilated every $30\Delta t$ and $\sigma_o^2 = 0.1$. The red dot-dash line represents the true solution \mathbf{x}^t , observations \mathbf{y} are given by circles, the background \mathbf{x}^b is given by the dashed blue line and the analysis \mathbf{x}^a is given by the solid green line.

the perfect observation case. With $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ the parameter estimates are very noisy. The size of the parameter errors increase as σ_o^2 increases and also as the frequency of the observations decreases. For observations up to every $20\Delta t$ the oscillations are approximately centered around the true parameter values. There is a significant growth in the size of the oscillations when the time between assimilations is increased to $30\Delta t$. If we try to extend the observation interval any further, the model fails to produce any meaningful results.

Figures 7.29 and 7.30 show the effect of averaging the parameter estimates over a moving time window. For observation intervals $5\Delta t$ to $20\Delta t$ the parameters were averaged over a moving window of 50 time steps, starting at $t = 10$. For observation interval $30\Delta t$ this was increased to 60 timesteps and averaging was started later at $t = 15$. With $\sigma_o^2 = 0.1$ the parameters ρ and β are predicted to a good level of accuracy, even when the observation interval is extended to $30\Delta t$. With $\sigma_o^2 = 0.25$ we get similarly good results for observation intervals of $5\Delta t$ to $20\Delta t$. When the observation frequency is decreased to every $30\Delta t$ the errors are much larger but the averaged estimates still offer a significant improvement on the unaveraged values.

The parameter σ estimates are less accurate for both $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$. When $\sigma_o^2 = 0.25$, the averaging produces only a slight improvement on the initial estimate of $\sigma = 11.0311$ for intervals greater than every $5\Delta t$. When observations are taken at $30\Delta t$ intervals the final σ estimate is actually further from the true value than the starting estimate. In the perfect observation experiments, we found that this parameter converged much slower than the other two. This perhaps suggests that the model is relatively insensitive to small deviations in its value. The averaged estimate could potentially be improved by starting the averaging at a later time.

For observation intervals of $5\Delta t$ to $20\Delta t$, the quality of the analysis for the state variables is good for both $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ even without the averaging of the parameter estimates. As figures 7.31 and 7.32 illustrate, there is a noticeable deterioration in the analysis when the observation frequency is decreased to $30\Delta t$, particularly in the $\sigma_o^2 = 0.25$ case. When $\sigma_o^2 = 0.1$, we found that the state analysis improved when the model was re-run using the time averaging of the parameter estimates as shown in figure 7.33. Here the model is quicker to stabilise and subsequently more closely follows the true

solution. We failed to see any substantial improvement when the same approach was tried for the case $\sigma_o^2 = 0.25$.

7.4 Summary

In this chapter we examined the effectiveness of our proposed new hybrid state-parameter data assimilation scheme. The method was applied to three simple dynamical system models, where the number of parameters increased with each model. Although each system has different characteristics the technique performed well in all three cases. As the results of our experiments illustrate, we were able to recover the true parameter values to a good level of accuracy, even when observational data were noisy. This had a positive impact on the skill of the forecast model and enabled more accurate predictions of the true model state.

As we would expect, there are limits to the success of the method; when observational data became too infrequent or too noisy, or if the initial state and parameter background estimates were particularly poor then we were unable to yield reliable results. The threshold for each model varied depending on properties the of model structure and the underlying dynamics but were not overly restrictive. We also found that we were less able to accurately identify parameters to which a model is relatively insensitive, as was the case for certain values in the oscillating system. This is not surprising as we cannot expect to be able to correct parameter errors that cause errors in the model solution that are on smaller scales than can be observed. Other parameter estimation techniques would also be likely to fail in such a scenario. This is linked to the concepts of observability and identifiability discussed in chapter 3. A parameter estimation method can only be expected to work reliably when both these properties hold. Future work will need to consider these issues in more depth and examine how they relate to our new hybrid approach.

In this work we assumed that the parameters in the oscillating and Lorenz models were uncorrelated and set the cross covariances between the parameters equal to zero. Whilst this assumption worked for these particular models it may not adequate for models in which the parameters are strongly correlated. A model sensitivity analysis can be used to

help identify the interdependence of parameters and ascertain whether cross correlations are needed. In this case, more attention will need to be given to the parameter error covariance matrix \mathbf{B}_{pp} and methods for defining the cross correlations will need to be considered.

The models we used here are of relatively low dimension and have only a small number of parameters. The increase in the dimension of the problem caused by the addition of the parameters to the state vector does not have a significant impact on the computational cost of the assimilation scheme and the re-calculation of the Jacobian matrix \mathbf{N}_k at each new assimilation time is not infeasible. We chose model discretisations that allowed us to obtain explicit expressions for the matrix \mathbf{N}_k thereby avoiding any additional computational complexity. We remark, however, that an explicit form for the Jacobian is not necessarily required. As we show in the next chapter, the matrix \mathbf{N}_k can, for example, be approximated using a simple local finite difference approach.

This study has provided a valuable insight into how the hybrid method is likely to perform in a range of dynamical systems. Overall, the approach has proved extremely successful and we are confident that it offers an efficient and versatile solution to the problem of approximating the state-parameter cross covariances. In the next chapter we apply our new algorithm to the 1D nonlinear sediment transport model introduced in chapter 4. This model presents further challenges as we encounter difficulties relating to parameter interdependence.

Chapter 8

1D sediment transport model

In this chapter we evaluate the potential for practical application of our new hybrid technique in a morphodynamic modelling context by testing the approach in an idealised sediment transport model.

Details of the model and its discretisation are given in sections 8.1 to 8.3. In section 8.4 we explain how the Jacobian matrix \mathbf{N}_k and hence the state-parameter cross covariances are approximated. The various elements of the experimental design are described in section 8.5. Results are presented in section 8.6 followed by a summary of the conclusions in section 8.7.

8.1 The model

We consider the 1D nonlinear sediment transport model introduced in chapter 4 and use a generic test case consisting of a smooth, initially symmetric, isolated bedform in an open channel. The bed level changes are governed by the sediment conservation equation (4.1)

$$\frac{\partial z}{\partial t} = - \left(\frac{1}{1 - \epsilon} \right) \frac{\partial q}{\partial x}. \quad (8.1)$$

Here $z(x, t)$ is the bathymetry or bed height, t is time, q is the total load sediment transport rate in the x direction and ϵ is the sediment porosity.

To calculate q we use the power law equation (4.2), but for ease of computation we assume that the current u takes only positive values. The sediment transport flux can then be written as

$$q = Au^n, \quad \text{with } u \geq 0, \quad (8.2)$$

where A and n are parameters whose values we wish to estimate.

One of the reasons for choosing this simplified scenario is that, under certain assumptions, it is possible to derive an approximate analytical solution to (8.1) and this will be useful for model validation purposes.

We assume that the amplitude of the bedform is sufficiently small relative to the water depth such that any variation in the elevation of the water surface can be ignored. The water height, h , can then be taken to be constant. We further assume that the water flux, F , is constant across the whole domain. The solution derived under these assumptions is only strictly valid when the migration speed of the bedform is slow relative to the flow velocity [43]. Given these assumptions, the depth averaged current can be rewritten in terms of the water height, flux and the bed height as

$$u = \frac{F}{h - z}. \quad (8.3)$$

This enables us to express the sediment transport rate q as a function of bed height z rather than u ,

$$q = A \left(\frac{F}{h - z} \right)^n. \quad (8.4)$$

Thus, q can be now differentiated with respect to z

$$\frac{\partial q}{\partial z} = AnF^n(h - z)^{-(n+1)}. \quad (8.5)$$

This allows us to rewrite the sediment conservation equation (8.1) in the quasi-linear advection form

$$\frac{\partial z}{\partial t} + c(z, q) \frac{\partial z}{\partial x} = 0, \quad (8.6)$$

where

$$c(z, q) = \left(\frac{1}{1 - \epsilon} \right) \frac{\partial q}{\partial z} \quad (8.7)$$

is the celerity of the bedform.

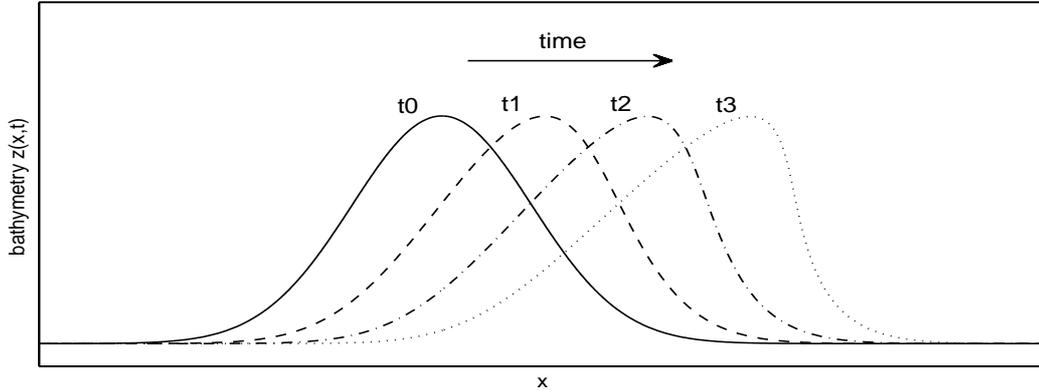


Figure 8.1: Schematic: solutions to the quasi-linear advection equation (8.6) for Gaussian initial data at times t_0 (solid line), t_1 (dashed line), t_2 (dot-dash line) and t_3 (dotted line), $t_0 < t_1 < t_2 < t_3$.

Using (8.5) we have

$$c(z) = \left(\frac{AnF^n}{1 - \epsilon} \right) (h - z)^{-(n+1)}, \quad (8.8)$$

i.e, the advection velocity of the bed is now a function of the bed height z only. As described in [88], equation (8.6) can be solved using the method of characteristics. The solution for an initial Gaussian hump is illustrated in figure 8.1. Since $c(z)$ is non-linear the top of the bedform moves faster than its base so that as the bed moves downstream it becomes distorted and eventually overturns. Whilst this type of solution would make sense in some contexts (such as a breaking wave) we would not expect it here. In real life, this overturning is prevented by natural phenomena such as bed-slope effects [94]. We create a more physically realistic solution that remains smooth and single valued by adding a small diffusive term to the right hand side of (8.6). Thus, the evolution of the bedform is now described by the nonlinear advection-diffusion equation

$$\frac{\partial z}{\partial t} + c(z) \frac{\partial z}{\partial x} = \kappa \frac{\partial^2 z}{\partial x^2}, \quad (8.9)$$

where κ is the diffusion coefficient.

8.2 Discretisation of the state evolution model

We discretise (8.9) using a hybrid semi-Lagrangian Crank-Nicolson (SLCN) algorithm based on that presented in [96]. The advection term (LHS of 8.9) is discretised using the

semi-Lagrangian technique as described in chapter 5 and this is then combined with the Crank-Nicolson methodology for the solution of diffusion problems. The scheme can be written as the matrix system

$$\left[I - \frac{\Delta t}{2} \mathcal{L} \right] \mathbf{z}_{k+1} = \mathcal{C}_* \left[\left(I + \frac{\Delta t}{2} \mathcal{L} \right) \mathbf{z}_k \right]. \quad (8.10)$$

The vectors \mathbf{z}_k and $\mathbf{z}_{k+1} \in \mathbb{R}^m$ represent the solution at times t_k and t_{k+1} on a discrete regular grid with spacing Δx and model time step Δt , where $x_j = j\Delta x$, $t_k = k\Delta t$. The operator $\mathcal{L} \in \mathbb{R}^{m \times m}$ is the discrete Laplacian operator corresponding to the second order spatial derivative and \mathcal{C}_* is an interpolation operator that interpolates the solution at time t_k from the regular model spatial mesh \mathbf{x} to the irregular departure points \mathbf{x}_d . For this work, we use a cubic spline interpolation.

The departure points are as defined in chapter 5 equation (5.11) for the linear advection model

$$\mathbf{x}_d = \mathbf{x} - \boldsymbol{\alpha}. \quad (8.11)$$

Here the vector \mathbf{x}_d contains the initial positions or departure points at time t_k of the particles arriving on the uniform model grid \mathbf{x} at time t_{k+1} and the vector $\boldsymbol{\alpha}$ represents the displacement of the particles in the x direction between t_k and t_{k+1} .

In chapter 5, the advection velocity c was constant and so the displacements, connecting the departure points \mathbf{x}_d at time t_k to the regular grid \mathbf{x} at time t_{k+1} , could be computed exactly. Here the advection velocity $c(z)$, given by equation (8.8), is a non-constant function of the bed height $z(x, t)$ and so evaluation of the displacements is less straightforward. We implement a two-time-level, semi-implicit algorithm based on that presented in [97]. The scheme uses a linear interpolation combined with an $\mathcal{O}(\Delta t^2)$ accurate extrapolation formula to iterately solve for the displacements and hence find \mathbf{x}_d . The Crank-Nicolson method uses a centered difference to approximate the second order spatial derivative; this gives the operator \mathcal{L} a particularly convenient structure. For zero boundary conditions, \mathcal{L} is given by the tri-diagonal matrix

$$\mathcal{L} = \begin{pmatrix} -2\nu & \nu & 0 & \dots \\ \nu & -2\nu & \nu & 0 \\ & \ddots & \ddots & \ddots \\ & & \ddots & \ddots \\ \dots & 0 & \nu & -2\nu \end{pmatrix}, \quad (8.12)$$

where $\nu = \frac{\kappa}{\Delta x^2}$.

The Crank-Nicolson scheme is second order accurate in both time and space and, like the semi-Lagrangian method, is unconditionally stable [71]. The Semi-Lagrangian and Crank Nicolson components of the scheme were tested and validated both independently and in combination. Details are given in appendix E.

Implementation of the semi-Lagrangian Crank-Nicolson algorithm (8.10) can be summarised as follows:

Algorithm 8.1 The semi-Lagrangian Crank Nicolson algorithm

- (i) Iteratively compute the vector of displacements $\boldsymbol{\alpha}$, for all grid points \mathbf{x} using values from the previous time step as an initial guess.
- (ii) Calculate the vector of departure points $\mathbf{x}_d = \mathbf{x} - \boldsymbol{\alpha}$.
- (iii) Apply the diffusion operator $(I + \frac{\Delta t}{2}\mathcal{L})$ to \mathbf{z}_k on the regular grid at time t_k and interpolate the resulting field to the departure points \mathbf{x}_d to obtain $\mathbf{z}_k(\mathbf{x}_d)$.
- (iv) Evaluate the solution \mathbf{z}_{k+1} on the regular grid \mathbf{x} at time t_{k+1} by solving the system (8.10) using a standard matrix equation solver.

8.3 The parameter model

The parameters A and n are constant, thus we have the standard discrete parameter evolution model

$$\mathbf{p}_{k+1} = \mathbf{p}_k, \quad k = 0, 1, \dots, T, \quad (8.13)$$

where

$$\mathbf{p}_k = \begin{pmatrix} A_k \\ n_k \end{pmatrix}. \quad (8.14)$$

The augmented system model is given by combining (8.14) with (8.10). The augmented state vector \mathbf{w}_k is a $m + 2$ vector where m is the dimension of the model state vector \mathbf{z}_k .

8.4 State-parameter cross covariances and parameter covariances

Unlike the models we considered in chapter 7, the Jacobian of the state forecast model with respect to the parameters cannot be easily computed from the model discretisation (8.10). A simple alternative is to use a local finite difference approximation.

Here, the Jacobian matrix \mathbf{N}_k is defined as

$$\mathbf{N}_k = \left(\begin{array}{cc} \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial A} & \frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial n} \end{array} \right) \Big|_{\mathbf{z}_k^a, \mathbf{p}_k^a}. \quad (8.15)$$

We approximate the derivatives $\frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial A}$ and $\frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial n}$ using a forward difference between the current model background forecast (made using the current model parameter estimate) and the forecast obtained by adding a small perturbation to the current parameter.

Defining

$$\mathbf{z}_{k+1}^b = \mathbf{f}(\mathbf{z}_k^a, \mathbf{p}_k^a) \quad \text{and} \quad \hat{\mathbf{z}}_{k+1}^b = \mathbf{f}(\mathbf{z}_k^a, \hat{\mathbf{p}}_k^a) \quad (8.16)$$

we approximate $\frac{\partial \mathbf{f}(\mathbf{z}, \mathbf{p})}{\partial A}$ as

$$\frac{\partial \mathbf{f}(\mathbf{z}_k^a, \mathbf{p}_k^a)}{\partial A} \approx \frac{\hat{\mathbf{z}}_{k+1}^b - \mathbf{z}_{k+1}^b}{\delta A} \quad (8.17)$$

where

$$\hat{\mathbf{p}}_k^a = \begin{pmatrix} A_k^a + \delta A \\ n_k^a \end{pmatrix}, \quad (8.18)$$

\mathbf{p}_k^a is the current parameter vector and δA is a small perturbation to the current estimate of A .

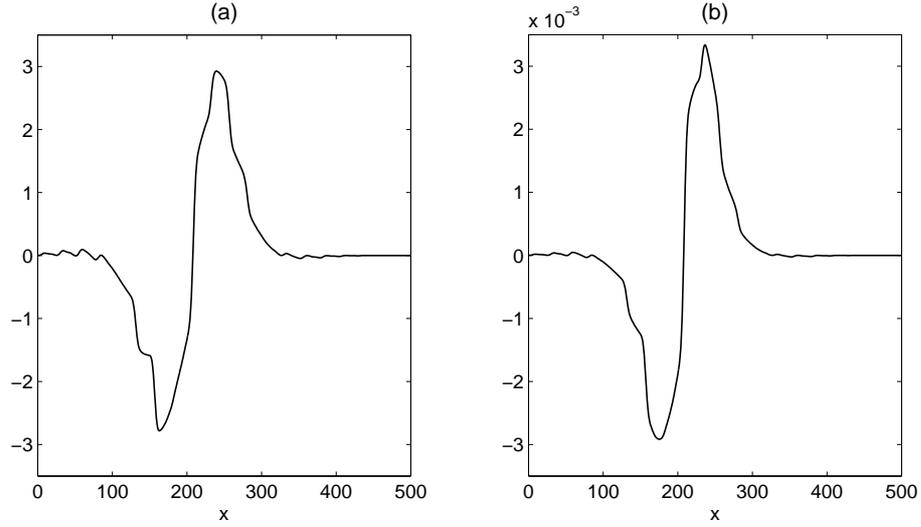


Figure 8.2: Approximating the matrix \mathbf{N}_k : (a) $\frac{\partial \mathbf{f}}{\partial A}$ and (b) $\frac{\partial \mathbf{f}}{\partial n}$ at $t = 1$ for an example assimilation run with $A_0 = 0.01 \text{ ms}^{-1}$, $\delta A = 10^{-5} \text{ ms}^{-1}$, $n_0 = 2.4$, $\delta n = 0.1$.

Similarly

$$\left. \frac{\partial \mathbf{f}}{\partial n} \right|_{\mathbf{z}_k^a, \mathbf{p}_k^a} \approx \frac{\hat{\mathbf{z}}_{k+1}^b - \mathbf{z}_{k+1}^b}{\delta n} \quad (8.19)$$

where

$$\hat{\mathbf{z}}_{k+1}^b = \mathbf{f}(\mathbf{z}_k^a, \hat{\mathbf{p}}_k^a) \quad \text{and} \quad \hat{\mathbf{p}}_k^a = \begin{pmatrix} A_k^a \\ n_k^a + \delta n \end{pmatrix}, \quad (8.20)$$

and δn is a small perturbation to the current approximation of n .

Figures 8.2 (a) and (b) show $\frac{\partial \mathbf{f}}{\partial A}$ and $\frac{\partial \mathbf{f}}{\partial n}$ calculated for an example assimilation run. The model is run forward over a single time step; once using the current parameter estimate (a) A_k^a or (b) n_k^a and a second time using the perturbed value $A_k^a + \delta A$ (or $n_k^a + \delta n$). The difference between the two forecasts divided by the perturbation is then computed thus giving an approximation to the model gradient (8.17) or (8.19).

Although the gradient approximations shown in figure 8.2 are not completely smooth this is due to variations in the smoothness of the background field rather than instability of the calculation. The accuracy and stability of the finite difference approximation will depend on the size of the parameter perturbation and the forecast time step. The choice of these values is related to the sensitivity of the model to changes in the parameters. A change in the value of A or n affects the model in similar ways. From equation (8.4) we know that both A and n affect the magnitude of the sediment transport rate q and this will in turn affect the bed celerity (8.8). A particular feature of our chosen flux formula

(8.4) is that the model is more sensitive to the parameter A than it is to n . Incorrect estimation of either parameter produces a phase error but the divergence of the model from the true solution is more rapid when the error is in A . If the current $u = \frac{F}{h-z}$ is close or equal to 1 ms^{-1} , a change in n will have little or no effect on the model.

Suitable values for δA and δn were determined by carrying out a model sensitivity study. The parameter perturbation must be sufficiently small that the behaviour of the model does not become unstable but sufficiently large so as to cause a measurable change. Additionally, if the forecast time step is too short it may not allow sufficient time for the model to react to the change in parameter value. Because the model is relatively insensitive to n , we found that a relatively large δn was required to produce a determinable change in the state forecast. The converse was true for A ; if δA was set too large the method became unstable. This type of approach to model sensitivity analysis can have limitations; we do not discuss this further here but note that many of the issues are neatly described in the paper by Errico [19].

A further problem with our chosen model formulation is that of equifinality or collinearity. This is linked to the issues of uniqueness and identifiability discussed in chapter 3 and occurs when there exists a range of different parameter combinations that produce similar model behaviour [72], [93]. This can make it difficult to distinguish individual parameters and can lead to estimates that are, strictly speaking, ‘incorrect’ but that are in practice sufficiently accurate when used to forecast the model over short time periods.

Initial experiments with this model indicated a strong interdependence between A and n . We found that when the state-parameter cross covariances are poorly specified, the tendency of the model is to compensate for errors in the value of n through A , i.e. an underestimated n value is offset by an overestimate of parameter A and vice versa. This highlights the need for greater consideration of the relationship between individual parameters and how this can be best described in terms of the covariance matrices \mathbf{B}_{zp} and \mathbf{B}_{pp} .

For the experiments with the oscillator and Lorenz models presented in the previous chapter, we assumed that the parameters were uncorrelated and set the off diagonal elements of the parameter error covariance matrix \mathbf{B}_{pp} equal to zero. Because of the

evident interaction between the parameters A and n and their errors, this assumption is not sufficient in this case.

We can write the covariance matrix of the errors in the parameter vector $\mathbf{B}_{\mathbf{pp}}$ as

$$\mathbf{B}_{\mathbf{pp}} = \begin{pmatrix} \sigma_A^2 & \sigma_{An} \\ \sigma_{nA} & \sigma_n^2 \end{pmatrix}, \quad (8.21)$$

where $\sigma_A^2 = E(\varepsilon_A^2)$ and $\sigma_n^2 = E(\varepsilon_n^2)$ are the error variances for A and n respectively and $\sigma_{An} = \sigma_{nA} = E(\varepsilon_A \varepsilon_n)$ is the covariance between the errors in A and n .

The state-parameter cross covariance matrix $\mathbf{B}_{\mathbf{zp}}$ is given by

$$\begin{aligned} \mathbf{B}_{\mathbf{zp}_{k+1}} &= \mathbf{N}_k \mathbf{B}_{\mathbf{pp}} \\ &= \begin{pmatrix} \frac{\partial \mathbf{f}}{\partial A_k} & \frac{\partial \mathbf{f}}{\partial n_k} \end{pmatrix} \begin{pmatrix} \sigma_A^2 & \sigma_{An} \\ \sigma_{An} & \sigma_n^2 \end{pmatrix} \\ &= \begin{pmatrix} \sigma_A^2 \frac{\partial \mathbf{f}}{\partial A_k} + \sigma_{An} \frac{\partial \mathbf{f}}{\partial n_k} & \sigma_n^2 \frac{\partial \mathbf{f}}{\partial n_k} + \sigma_{An} \frac{\partial \mathbf{f}}{\partial A_k} \end{pmatrix} \end{aligned} \quad (8.22)$$

The question remains as to how to approximate the parameter covariance σ_{An} ; this will be discussed in the results section 8.6 in the context of specific examples.

8.5 Assimilation experiments

We test our scheme by running a series of identical twin experiments. For the purpose of these experiments we assume that the values of h , F and ϵ are known and constant but that the true values of the parameters A and n are uncertain. The water height and flux are specified as $h = 10.0$ m, $F = 7.0$ m and the sediment porosity is given the default value $\epsilon = 0.4$. The diffusion coefficient is set at $\kappa = 0.001 \text{ m}^2\text{s}^{-1}$; we found that this value was sufficient to prevent the bed from overturning without overly smoothing the solution.

We assume that our numerical model is perfect and generate a reference or ‘true’ solution by running the model on the domain $x \in [0, 500]$ with grid spacing $\Delta x = 1.0$ m, time

step $\Delta t = 30$ minutes and parameter values $A = 0.002 \text{ ms}^{-1}$ and $n = 3.4$. The true initial bathymetry is taken as

$$z(x, 0) = \begin{cases} 0 & x < 50 \\ e^{-\frac{(x-200)^2}{2 \times 30^2}} & 50 < x < 450 \\ 0 & x \geq 450 \end{cases} \quad (8.23)$$

This solution is used to provide pseudo-observations for the data assimilation and also to evaluate the performance of our scheme. The model is then re-run with the data assimilation, starting from a perturbed initial bathymetry and with incorrect starting values for the parameters A and n .

The initial model background state is of the form (8.23) but with slightly different scaling factors. The state background error covariance matrix \mathbf{B}_{zz} is given by the isotropic correlation function (5.23). The correlation length scale L is adjusted empirically as described in chapter 5. For these experiments a value of four times the current observation spacing was found to work well. The background error variance is set at $\sigma_b^2 = 0.05$.

For the parameter error covariance matrix, we set σ_A^2 and σ_n^2 equal to the square of the initial error; the parameter cross covariance σ_{An}^2 is specified according to the direction of these errors, as described in the following results section.

The state-parameter cross covariance matrix \mathbf{B}_{zp_k} is recalculated at each new assimilation time as described in section 8.4 with perturbations $\delta A = 10^{-5} \text{ ms}^{-1}$ and $\delta n = 10^{-1}$.

Observations of the bed height z are taken directly from the reference solution and are assimilated sequentially at regular time intervals. The space and time frequency of the observations is kept fixed for each individual experiment but is varied between experiments as discussed in the results section below. We use a constant diagonal matrix for the observation error covariance matrix

$$\mathbf{R}_k = \mathbf{R} = \sigma_o^2 \mathbf{I}, \quad \mathbf{I} \in \mathbb{R}^{r \times r}, \quad (8.24)$$

where r is the number of observations.

Experiments were carried out using both perfect and noisy observations. For the perfect observation experiments we set the observation error variance $\sigma_o^2 = 0.01$; for the experiments with noisy observation σ_o^2 is set equal to the variance of the added noise.

The augmented cost function is minimised iteratively using the quasi-Newton descent algorithm presented in section (5.2). At the end of each assimilation cycle the values of A and n are updated and the state analysis is integrated forward using the model (with the new parameter values) to become the background state for the next assimilation time.

8.6 Results

8.6.1 Perfect observations

Experiments were repeated for a range of starting combinations of A and n and for a range of time and space steps. Figures 8.3 and 8.4 show the parameter A and n updates for two example model runs with initial parameter estimates (i) $A_0 = 0.01 \text{ ms}^{-1}$ (overestimated), $n_0 = 2.4$ (underestimated) and (ii) $A_0 = 0.0 \text{ ms}^{-1}$ (underestimated), $n_0 = 4.4$ (overestimated). Here observations are taken at fixed space intervals of $25\Delta x$ and assimilated at time intervals varying from every $2\Delta t$ (1 hour) to $24\Delta t$ (12 hours). The parameter cross covariance is assumed to be strongly negative. The speed of convergence of the estimates decreases as the time frequency of the observations decreases but the scheme retrieves the true values of A and n to a high level of accuracy in both cases. We found that when A is initially underestimated (as in example (ii)) the convergence of the estimates was much slower than when A is initially overestimated. This is believed to be due to our model being much more sensitive to the parameter A than it is to the parameter n . Here, a low A estimate combined with a good approximation of the initial bathymetry means that the model predicted bathymetry does not diverge away from the true bathymetry as rapidly as when A is overestimated. Because the time between successive assimilations is short and we are weighting towards the observations, the difference between the modelled and true bathymetry remains small. The observation minus background increments are therefore small, leading to small analysis increments and hence slower updating. In other words, the state estimation alone is good enough to compensate for the incorrect parameters over short timescales. We found that the convergence of the estimates could be improved by inflating the error variances. The parameter updates illustrated in figure 8.4 were obtained by increasing

the estimated parameter error variances by a factor of 20. The estimates obtained using the non-inflated error variances are not shown.

When the initial overestimation of A is relatively large we get the opposite effect. For example (i) (figure 8.3) we see much greater fluctuation in the parameter values as the time frequency of the observations is decreased. The state forecast model reacts to these changes and consequently takes longer to stabilise as is illustrated in figure 8.6. Once the model has settled the quality of the analysis for the bathymetry z is excellent.

Figure 8.5 shows the effect of varying the grid spacing between observations from every $10\Delta x$ to every $100\Delta x$ for example (i). The observations are assimilated at fixed time intervals of $4\Delta t$ (2 hours). The effect is similar to decreasing the time frequency with the speed of convergence decreasing as the number of observations is decreased. Again, the scheme proves effective at recovering the true parameter values to high accuracy. For observation intervals up to $50\Delta x$ the quality of the analysis for z is consistently good and so we do not present the results here. As figure 8.7 illustrates, when observations are taken every $100\Delta x$ the height and shape of the predicted bathymetry varies depending on the positioning of the observations relative to the main body of the sediment; most notably whether an observation is taken sufficiently close to the peak of the curve.

8.6.2 Discussion

In these examples we have chosen our initial estimates of A and n such that the initial parameter errors ε_A and ε_n are of opposite sign. When A is over (under) estimated the increments in A need to be negative (positive), and the same applies for n . Both the state and parameters are updated according to the observations, exactly how the information in these observations is used depends on the background error covariances. Specifically, the magnitude and direction of the parameter updates will depend on the magnitude and direction of the state-parameter cross covariances \mathbf{B}_{zA} and \mathbf{B}_{zn} and these in turn depend on the error variances σ_A^2 , σ_n^2 and cross covariance σ_{An} . To ensure that the parameters are updated correctly, our choice of σ_A^2 , σ_n^2 and σ_{An} needs to be consistent with the true error statistics.

We know that A and n effect the model in similar ways. Figure 8.2 in section 8.4 shows

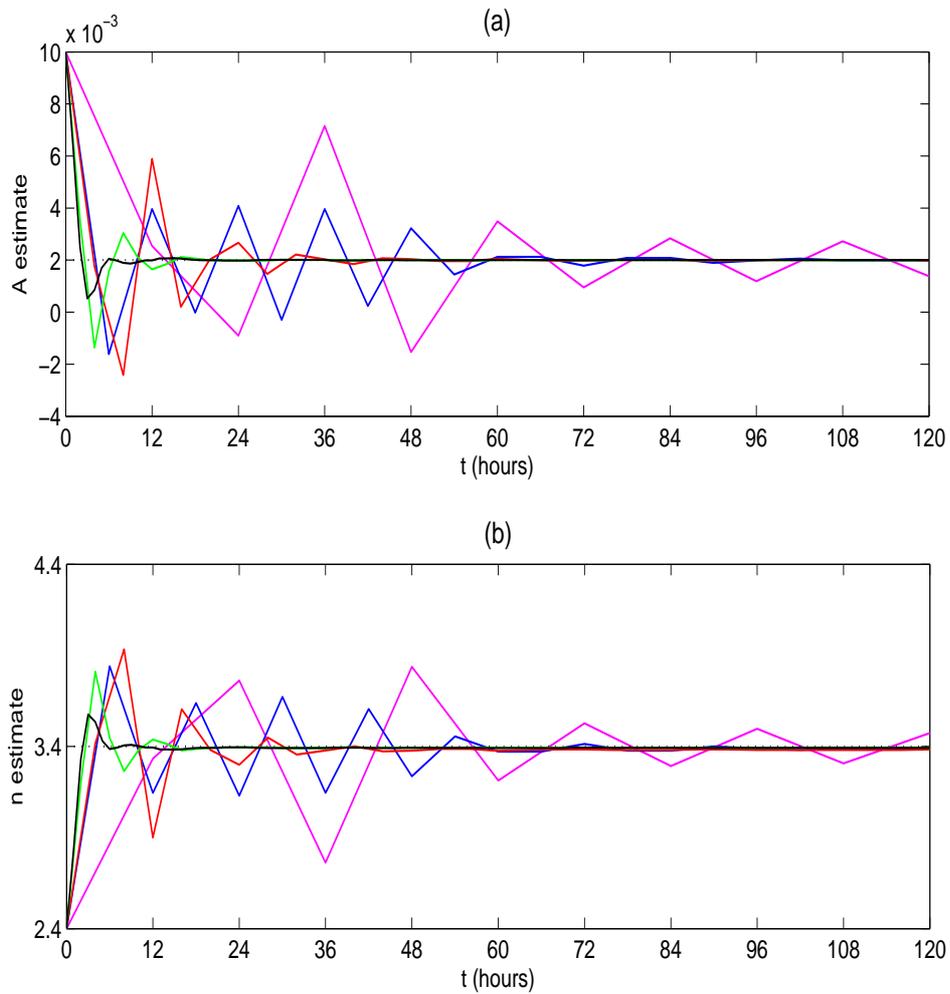


Figure 8.3: 1D sediment transport model: varying the time frequency of observations. Parameter updates for initial estimates $A_0 = 0.01 \text{ ms}^{-1}$, $n_0 = 2.4$. **(a)** estimated A values against time, **(b)** estimated n values against time. Solid black line - observations every $2\Delta t$; solid green line - observations every $4\Delta t$; solid red line - observations every $8\Delta t$; solid blue line - observations every $12\Delta t$; solid pink line - observations every $24\Delta t$. The true parameter values are indicated by the dotted black line.

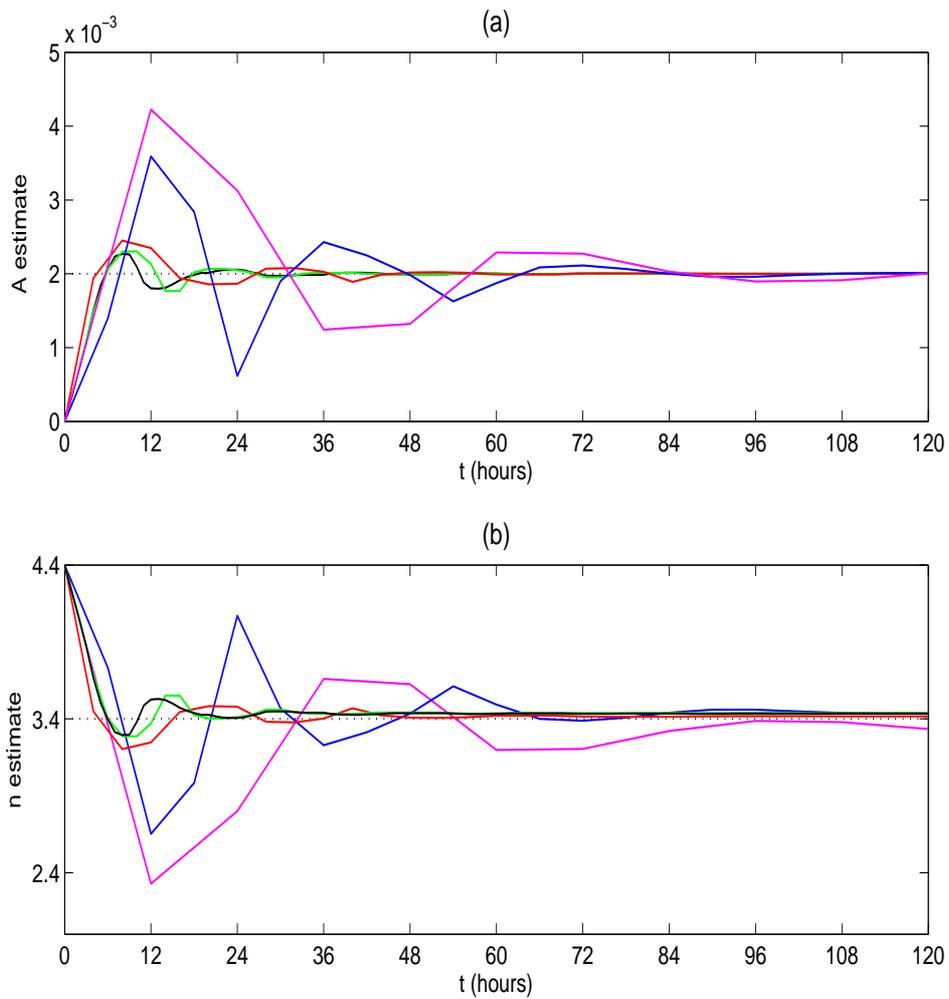


Figure 8.4: 1D sediment transport model: varying the time frequency of observations. Parameter updates for initial estimates $A_0 = 0.0 \text{ ms}^{-1}$, $n_0 = 4.4$. **(a)** estimated A values against time, **(b)** estimated n values against time. Solid black line - observations every $2\Delta t$; solid green line - observations every $4\Delta t$; solid red line - observations every $8\Delta t$; solid blue line - observations every $12\Delta t$; solid pink line - observations every $24\Delta t$. The true parameter values are indicated by the dotted black line.

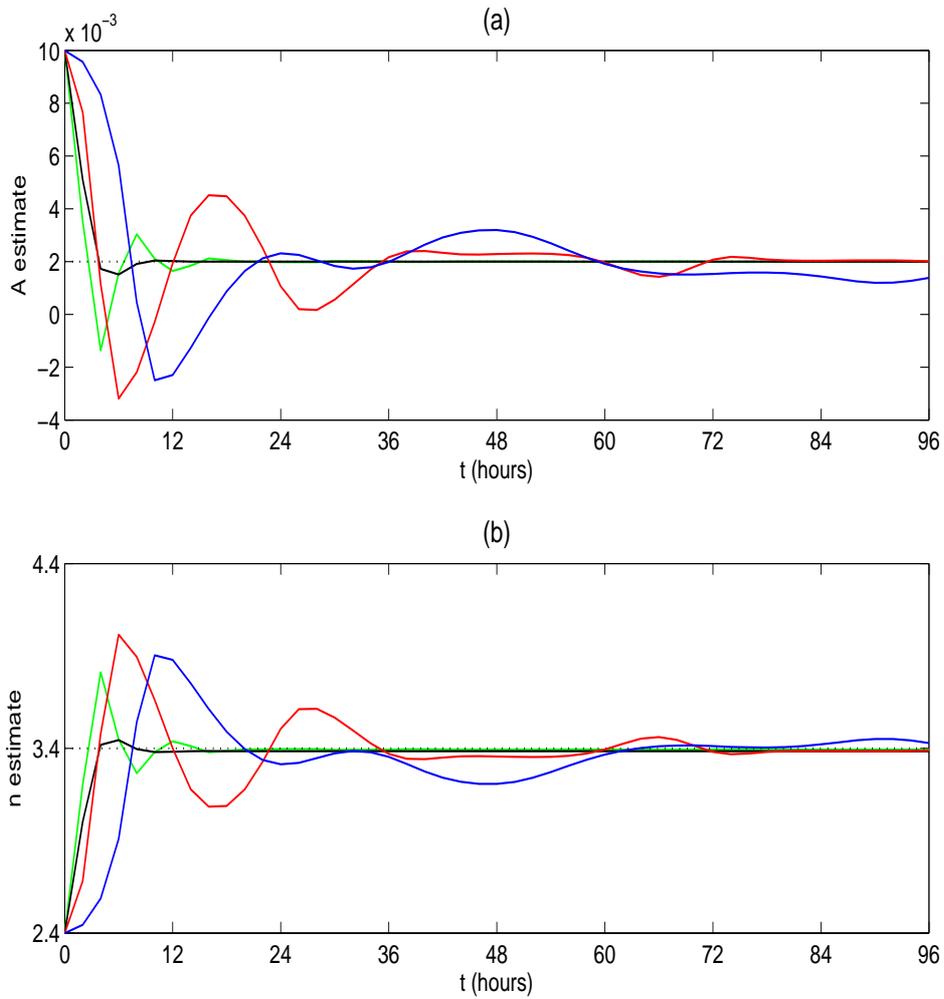


Figure 8.5: 1D sediment transport model: varying the spatial frequency of observations. Parameter updates for initial estimates $A_0 = 0.01 \text{ ms}^{-1}$, $n_0 = 2.4$. **(a)** estimated A values against time, **(b)** estimated n values against time. Solid black line - observations every $10\Delta x$; solid green line - observations every $25\Delta x$; solid red line - observations every $50\Delta x$; solid blue line - observations every $100\Delta x$. The true parameter values are indicated by the dotted black line.

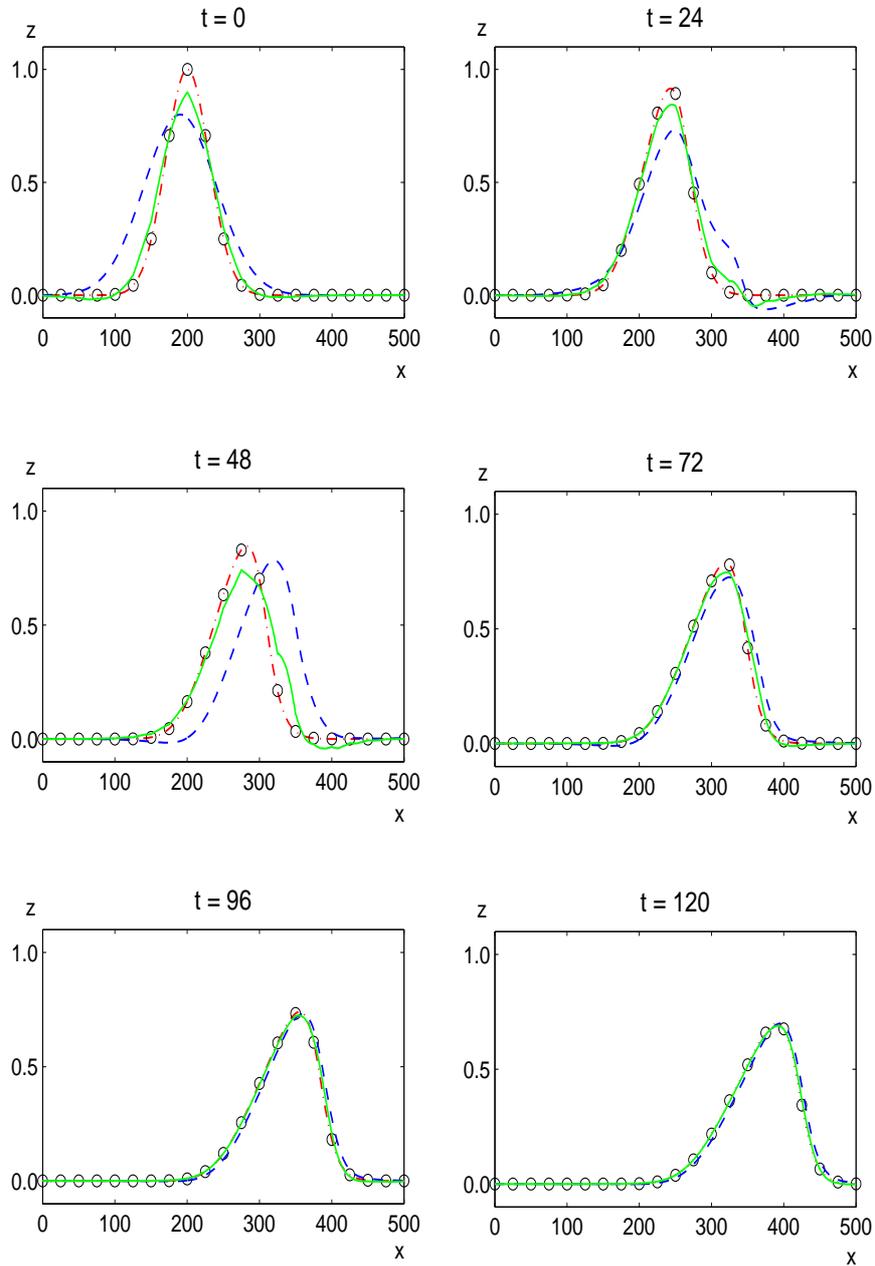


Figure 8.6: 1D sediment transport model: state analysis at times $t = 0, 24, 48, 72, 96$ & 120 , for initial parameter estimates $A_0 = 0.01 \text{ ms}^{-1}$, $n_0 = 2.4$ with perfect observations every $25\Delta x$, $4\Delta t$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

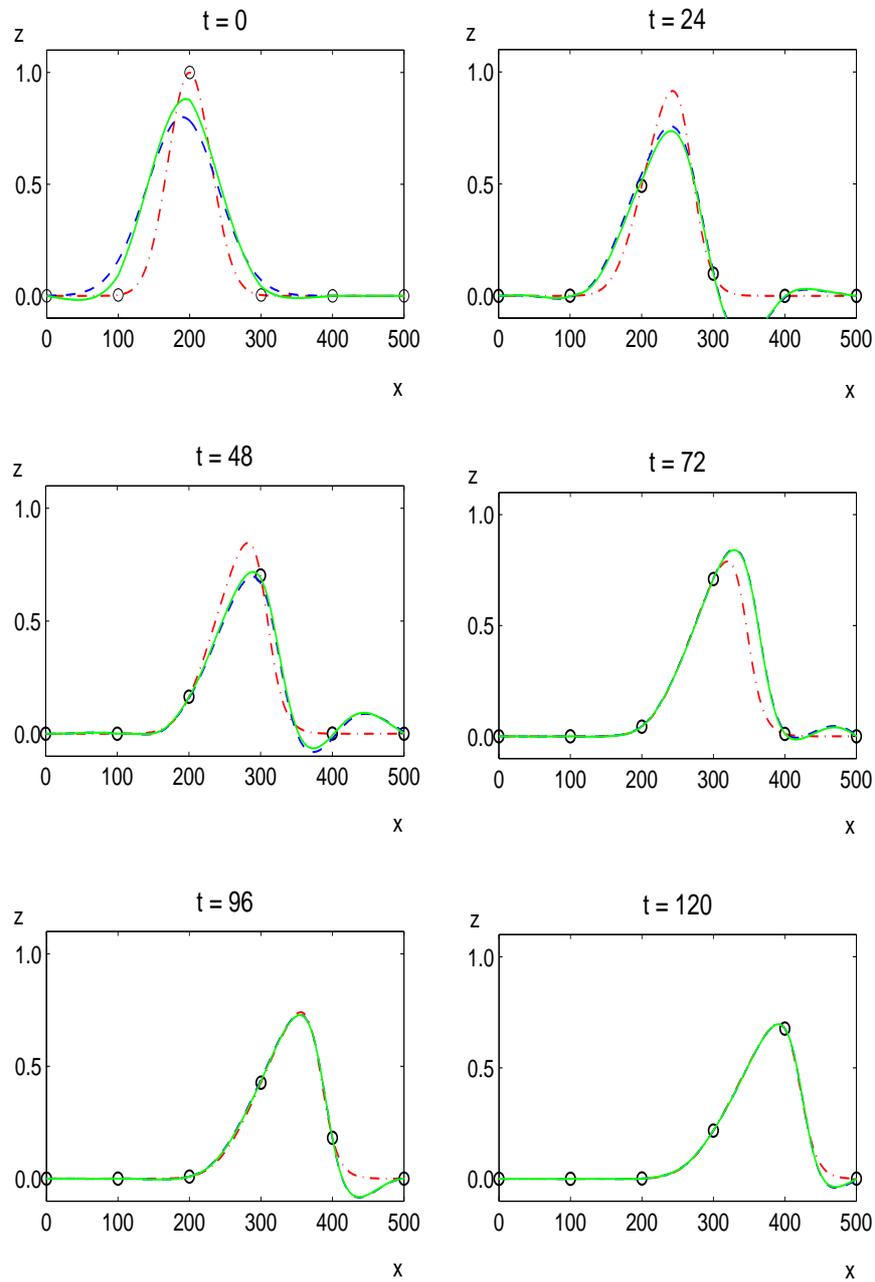


Figure 8.7: 1D sediment transport model: state analysis at times $t = 0, 24, 48, 72, 96$ & 120 , for initial parameter estimates $A_0 = 0.01 \text{ ms}^{-1}$, $n_0 = 2.4$ with perfect observations every $100\Delta x$, $4\Delta t$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

that $\frac{\partial \mathbf{f}}{\partial A}$ and $\frac{\partial \mathbf{f}}{\partial n}$ have a close resemblance and are generally expected to take the same sign. From (8.22), we have that if the parameter cross covariance $\sigma_{An} \geq 0$ then the elements of \mathbf{B}_{zA} and \mathbf{B}_{zn} will have the same sign and so A and n will both be updated in the same direction. When the errors in A and n are in opposite directions we need \mathbf{B}_{zA} and \mathbf{B}_{zn} to take opposite signs. We can achieve this by setting $\sigma_{An} < 0$ and choosing a σ_A^2 and σ_n^2 that take account of the relative magnitude of the errors in the parameter estimates.

Figure 8.8 illustrates the effect of setting $\sigma_{An} > 0$ in example (i) $A_0 = 0.01 \text{ ms}^{-1}$ (overestimated) and $n_0 = 2.4$ (underestimated). We find that the parameters A and n are now both updated in the same direction. When this experiment was repeated for other parameter combinations it was found that the direction of the parameter increments is dominated by the direction of the error in the A estimate, resulting in the parameter n being updated in the wrong direction. This is most likely due to our model being more sensitive to small changes in A than in n . The A estimates converge towards the true value of A but are generally less accurate than when σ_{An} is correctly specified, possibly due to the model compensating for the error in n . Despite the inability to recover the correct n value, the analysis for the bathymetry z is extremely accurate.

8.6.3 Noisy observations

The impact of observation error was examined by adding random Gaussian noise to the observations. Figure 8.9 shows the parameter updates produced for initial parameter estimates $A_0 = 0.0 \text{ ms}^{-1}$ (underestimated) and $n_0 = 4.4$ (overestimated) with error variances $\sigma_o^2 = 0.001$, $\sigma_o^2 = 0.01$ and $\sigma_o^2 = 0.1$. An error variance $\sigma_o^2 = 0.1$ is equivalent to 10% of the maximum bed height and is believed to represent a realistic level of measurement error. For these examples, observations were taken at $25\Delta x$ intervals and assimilated every $4\Delta t$ (2 hours). When $\sigma_o^2 = 0.001$ and $\sigma_o^2 = 0.01$ the parameter estimates fluctuate back and forth about their true values. The response is much slower when $\sigma_o^2 = 0.1$. We would expect the amplitude of the oscillations to increase as σ_o^2 is increased but here we find that the updates are actually less erratic. For these experiments the background error variance was kept fixed at $\sigma_b^2 = 0.05$; as σ_o^2 is increased the relative weighting of the background increases. When $\sigma_o^2 = 0.1$ we are actually weighting

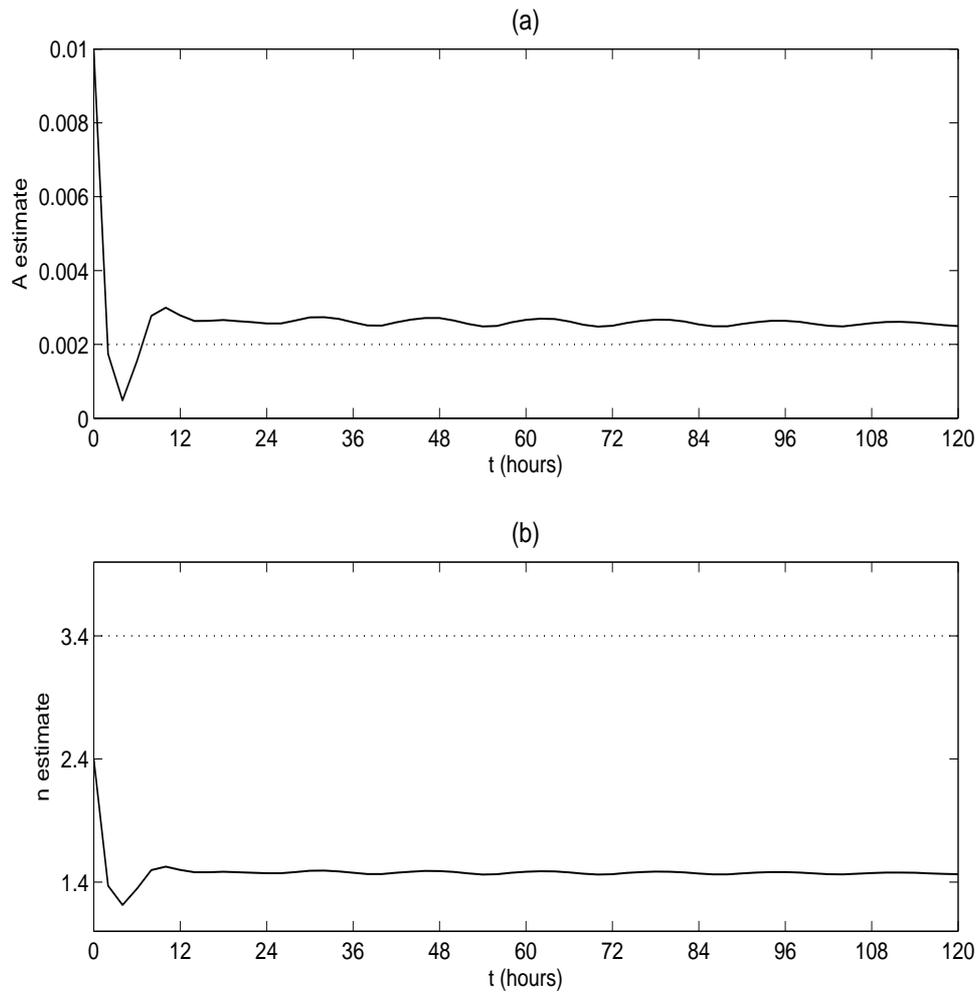


Figure 8.8: 1D sediment transport model: incorrect parameter cross covariance. Parameter updates for initial estimates $A_0 = 0.01 \text{ ms}^{-1}$, $n_0 = 2.4$ with observations every $25\Delta x$, $4\Delta t$. **(a)** estimated A values against time, **(b)** estimated n values against time. The true parameter values are indicated by the dotted black line.

towards the background state and this has the effect of smoothing the updates. Similar results were observed in a set of related experiments discussed in [91].

In all cases, the parameter estimates could be improved by averaging over a moving time window, as illustrated in figure 8.10. In the examples shown, we use a time window of 12 time steps. When $\sigma_o^2 = 0.001$ and $\sigma_o^2 = 0.01$ averaging is started at $t = 24$ hours. This is increased to $t = 48$ hours when $\sigma_o^2 = 0.1$ to allow for the fact that the estimates are converging more slowly.

In terms of the predicted bathymetry z , there are only very small differences between the truth and the model when the observation errors have variances $\sigma_o^2 = 0.001$ and $\sigma_o^2 = 0.01$. When σ_o^2 is increased to $\sigma_o^2 = 0.1$ there is a noticeable deterioration in the shape of the bed profile, as illustrated in figure 8.11. However, it is worth noting that the model still manages to track the position of the solution with reasonable accuracy.

8.7 Summary

In this chapter we applied our new hybrid assimilation method in an idealised 1D sediment transport model. There are two significant differences between this and the models we used in chapter 7. Firstly, the model has two uncertain, but highly correlated parameters. This means that in addition to the state-parameter cross covariances being well defined, it is also important that the parameter error covariances (as described by the matrix \mathbf{B}_{pp}) are given a good a priori specification. Because of the strong correlation between A and n , the values assigned to the parameter error variances σ_A^2 and σ_n^2 and cross covariance σ_{An} has a significant effect on the accuracy of the estimates obtained. To ensure that the scheme produces updates that are in the right direction and consistent with the scales of the model, σ_{An} should be non-zero and most crucially take the correct sign. We know from the results of the experiments in chapter 7 that this is less important for applications without parameter interdependence, or where parameters are only weakly correlated and therefore more easily identifiable.

The importance of the parameter correlations will also depend to a certain extent on the purpose of the parameter estimation. In some situations, the values of the individual

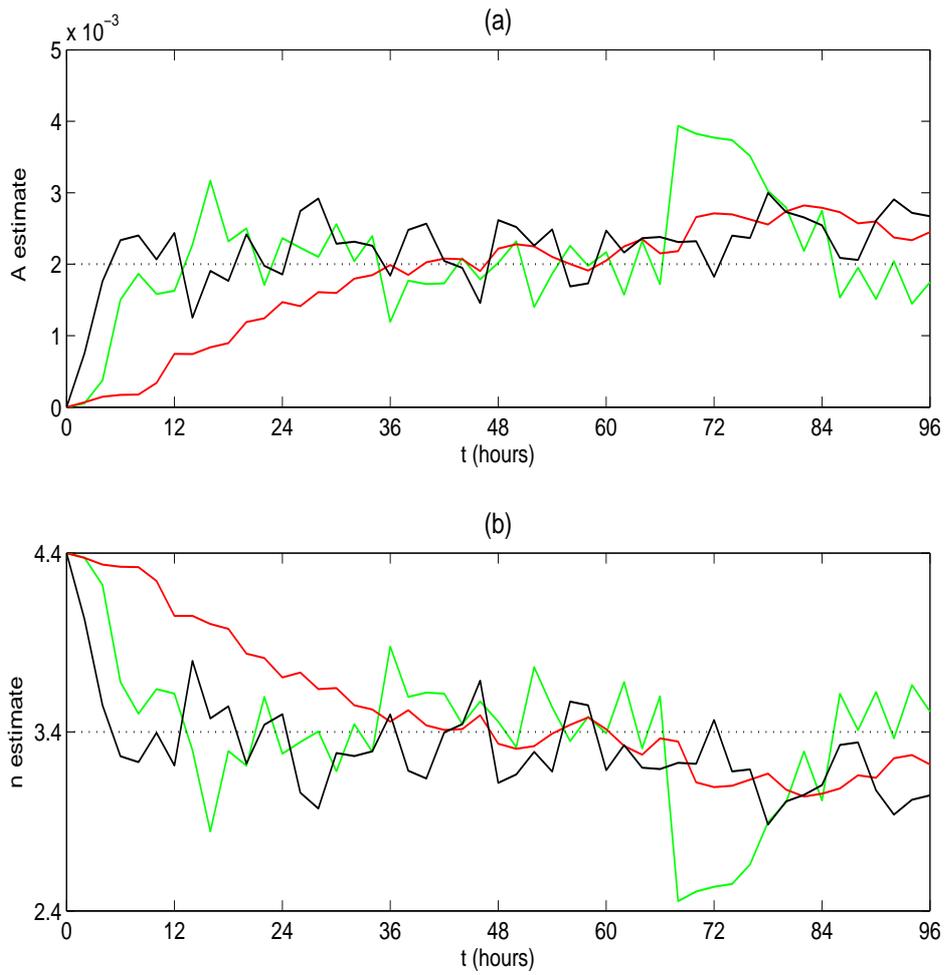


Figure 8.9: 1D sediment transport model: imperfect observations. Parameter updates for initial estimates $A_0 = 0.0 \text{ ms}^{-1}$, $n_0 = 4.4$. **(a)** estimated A values against time, **(b)** estimated n values against time. Solid black line - $\sigma_o^2 = 0.001$; solid green line - $\sigma_o^2 = 0.01$; solid red line - $\sigma_o^2 = 0.1$. The true parameter values are indicated by the dotted black line.

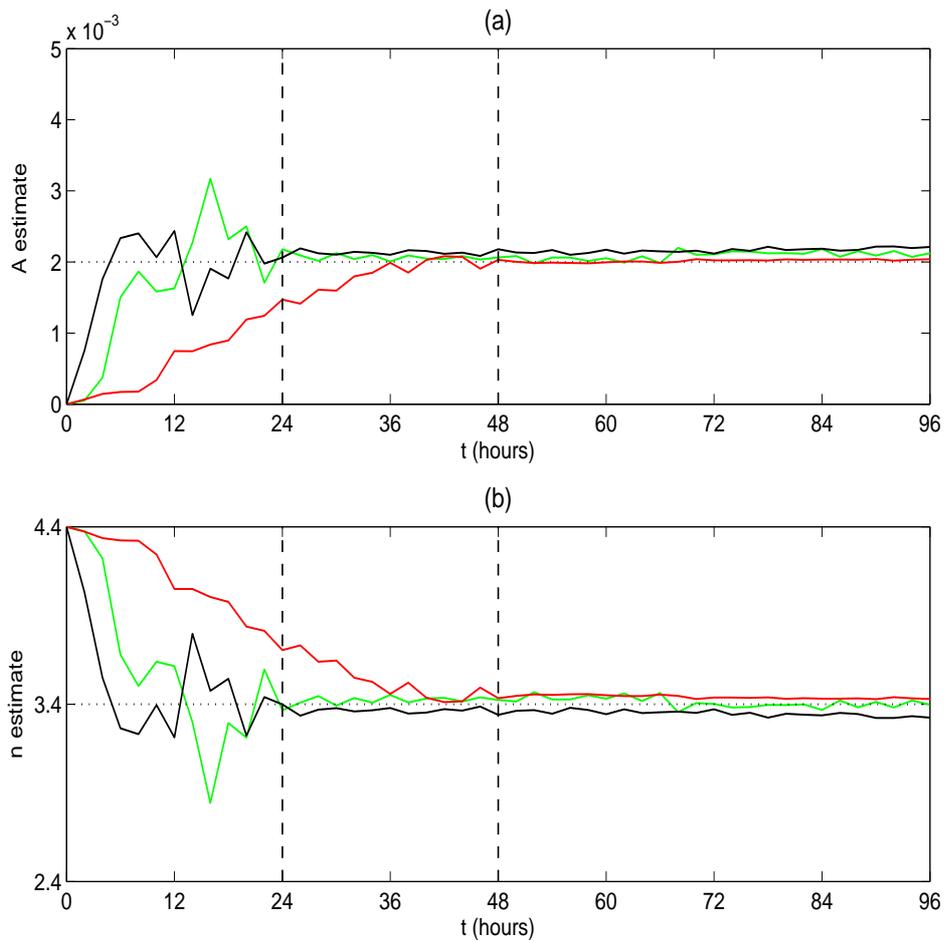


Figure 8.10: 1D sediment transport model: imperfect observations. Averaged parameter updates for initial estimates $A_0 = 0.0 \text{ ms}^{-1}$, $n_0 = 4.4$. **(a)** estimated A values against time, **(b)** estimated n values against time. Solid black line - $\sigma_o^2 = 0.001$; solid green line - $\sigma_o^2 = 0.01$; solid red line - $\sigma_o^2 = 0.1$. The true parameter values are indicated by the dotted black line. Averaging is started at $t = 24$ when $\sigma_o^2 = 0.001$, $\sigma_o^2 = 0.01$ and $t = 48$ for $\sigma_o^2 = 0.1$.

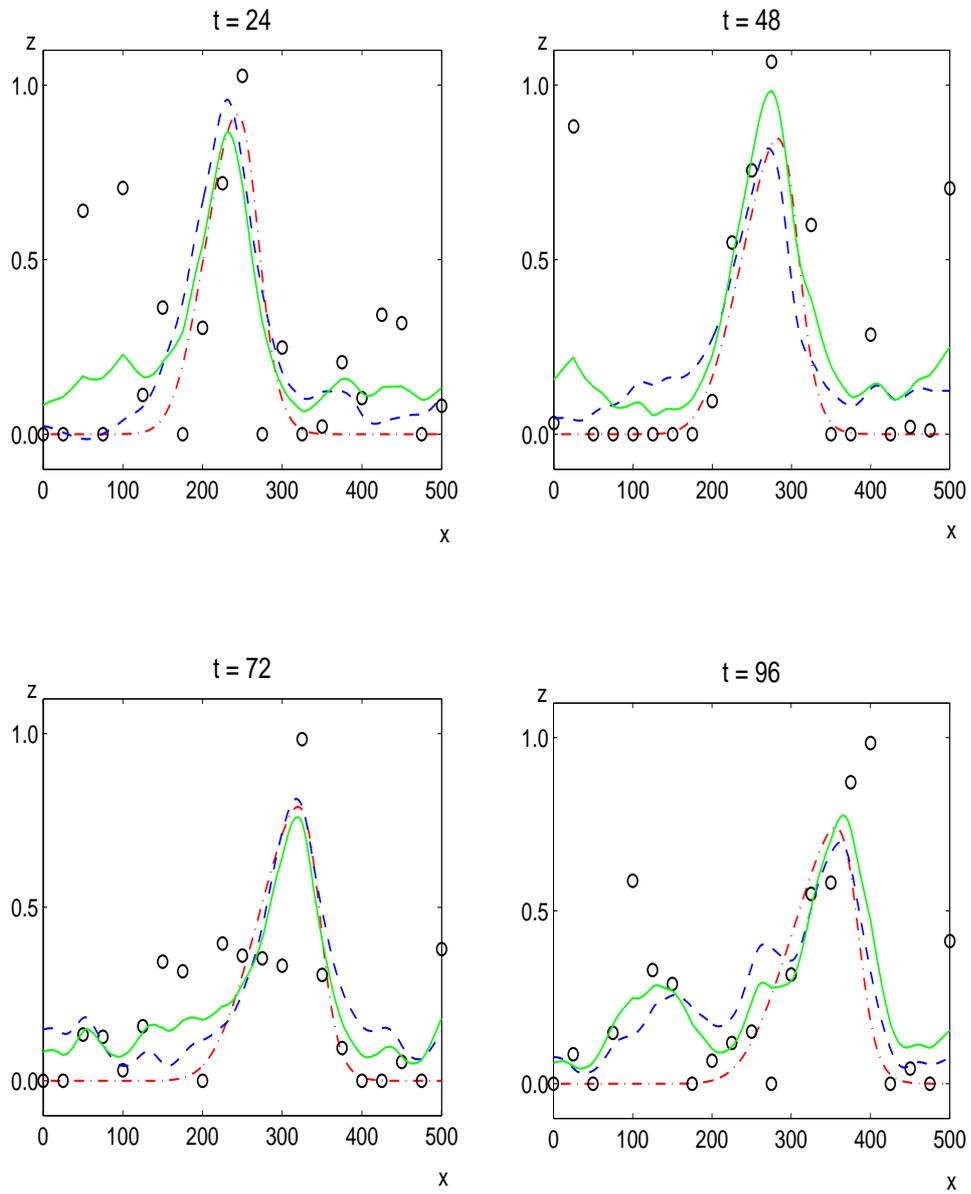


Figure 8.11: 1D sediment transport model: state analysis at times $t = 24, 48, 72$ & 96 , for initial parameter estimates $A_0 = 0.0 \text{ ms}^{-1}$, $n_0 = 4.4$ with noisy observations every $25\Delta x$, $4\Delta t$ and $\sigma_o^2 = 0.1$. The red dot-dash line represents the true solution \mathbf{z}^t , observations \mathbf{y} are given by circles, the background \mathbf{z}^b is given by the dashed blue line and the analysis \mathbf{z}^a is given by the solid green line.

parameters themselves may not be important. For example, we may simply be concerned with finding a set of parameters that give state estimates with a good fit to observations. As long as the future behaviour of the model with the parameters in combination is sufficiently accurate it may not matter that the individual parameter values are incorrect. In such cases the prescription of the parameter correlations will be less critical. In other circumstances, it may be that we want to identify individual parameter values in order to help better understand the underlying processes of a system or because we want to use the parameters in another context.

In practical situations, where the true statistics of the errors are not known, a model sensitivity analysis could be used to help identify the interdependence of parameters and ascertain whether cross correlations are needed. Here the correlation between the parameters A and n was caused by the formulation of the model. In other such cases, it may be prudent to consider a reparameterisation of the model equations to improve the identifiability of the parameters. Another possible approach is to transform the parameters to a set of uncorrelated variables [93].

The second main difference was the way in which the Jacobian matrix \mathbf{N}_k was approximated. The model discretisation was such that an explicit computational form for \mathbf{N}_k was not available. Instead, \mathbf{N}_k was approximated by calculating a local finite difference approximation of the gradient of the model with respect to the parameters. Because the number of parameters is small and the dimension of the state vector is relatively low this calculation does not add a significant amount to the computational cost of the approach and is therefore not impracticable. There could, however, potentially be issues if an efficient means of approximating \mathbf{N}_k is not available and/ or the state vector and the number of parameters to be estimated is large.

Despite the additional complexities presented by this model our hybrid scheme proved extremely effective. The results of our experiments have further demonstrated the versatility of the approach and strengthening the expectation that there is potential for successful application of this new methodology to larger, more realistic models with more complex parameterisations.

Chapter 9

Conclusions and further work

Parameters are inherent to numerical models. Specification of model parameters is a critical part of model development; even with perfect initial data, inaccurate representation of model parameters will lead to the growth of model error and therefore affect the ability of a model to accurately predict the true system state. In this thesis we sought to address the problem of uncertain model parameter estimation using data assimilation techniques in the context of morphodynamic modelling. We now conclude with a summary of our results and a discussion of some ideas for future research.

9.1 Conclusions

Data assimilation is predominantly used for state estimation so we began, in chapter 2, by describing the data assimilation problem for state estimation in a general system model.

Generally, parameter estimation is addressed as a separate issue to state estimation and model calibration is performed offline in a separate calculation. The classic approach is manual calibration or tuning of the model against observational data, but a variety of other approaches have been developed. In chapter 3 we discussed some of the different methods used for the estimation of parameters in coastal modelling applications along with their strengths and weaknesses. We then introduced the technique of state aug-

mentation and explained how it is possible to use data assimilation to estimate uncertain model parameters jointly with the model state variables. The state augmentation framework allows us to estimate the poorly known model parameters whilst simultaneously updating the predicted model state, rather than treating state and parameter estimation as two individual processes. The approach enables more efficient state and parameter estimation; making better use of the available observational data and potentially saving on calibration time. Data assimilation techniques have the further advantage over many other parameter estimation methods in that they offer a framework for explicitly accounting for all sources of uncertainty.

In principle, state augmentation can be applied with any of the standard data assimilation methods. Here we considered application of the technique to sequential 3D-Var data assimilation. Variational data assimilation is a popular choice for state estimation in large problems. The 3D-Var approach was primarily chosen for its computational robustness and ease of implementation. By adopting a sequential approach we are able to utilise new observations as they become available.

In chapter 5 we used a simple single parameter 1D linear advection model to help demonstrate the theory and highlight some of the issues associated with practical implementation of the 3D-Var and state augmentation techniques. A key difficulty in the construction of a data assimilation algorithm is specification of the background error statistics. These statistics, in the form of error covariances, govern the spatial spreading and smoothing of the observational data and therefore play a fundamental role in determining the nature of the analysis. Generally, the assumption in 3D-Var is that the statistics of the background errors are homogeneous, isotropic, and independent of the flow. We demonstrated the effect the choice of state background error covariance model can have on the quality of the state analysis via a series of simple state estimation experiments. We found that overall a full, symmetric matrix constructed using a Gaussian function produced the best results. We were able to use the correlation length scale L to control the amount of information smoothing.

For joint state-parameter estimation the background error covariance matrix must also include a description of the covariances of the parameter errors and the cross covariances between the state and parameter errors. Since it is not possible to observe the

parameters themselves, the parameter estimates depend on the observations of the state variables. It is the state-parameter cross covariances that pass information from the observed variables to update the estimates of the unobserved parameters. These covariances therefore play a crucial role in the parameter updating. The success of the state augmentation approach relies on the relationship between the state and parameters being well defined and assumes that we have sufficient knowledge to reliably prescribe the a priori state-parameter cross-covariances.

We considered ways of modelling the state-parameter cross covariances within the context of our simple linear advection model (chapter 5) by examining form of the background errors under the assumption that both the initial state and parameter are unknown. We initially applied the same principle as for the state background error covariance matrix, and prescribed the state-parameter cross covariances a static functional form but this failed to produce reliable estimates. We then investigated using a flow dependent structure; this was found to work extremely well, with the scheme producing accurate estimates of both the model state and parameters.

Importantly, the results of our experiments showed that it is not necessary to explicitly propagate the full augmented background error covariance matrix. We were able to get accurate estimates of both the parameters and state variables by combining a time varying approximation of the state-parameter cross covariance matrix \mathbf{B}_{zp_k} with an empirical, static representation of the state background error covariance \mathbf{B}_{zz} and static parameter error covariance matrix \mathbf{B}_{pp} .

In chapter 6 we considered the question of how to specify the augmented background error covariance matrix for a general system model. We used the results of chapter 5 to establish a novel approach that is applicable to a range of dynamical system models. Our new hybrid algorithm combines ideas from 3D-Var and the EKF and enables us to capture the flow dependent nature of the state-parameter cross covariances without the computational expense of explicitly propagating the full system covariance matrix. A simplified version of the EKF forecast step is used to estimate the state-parameter forecast error cross covariances and this is then combined with an empirical, static approximation of the state background error covariances and a static parameter error covariance matrix.

The method is relatively easy to implement and computationally inexpensive to run. The updating of the state-parameter cross covariances involves computing an approximation of the Jacobian of the state forecast model with respect to the parameters, but since the number of parameters to be estimated is typically quite small this does not add significantly to the overall cost of the assimilation scheme.

Key advantages of the hybrid approach are that the background error covariance matrix only needs to be updated at each new analysis time rather than at every time step and it does not require the previous cross covariance matrices to be stored. It also avoids many of the potential problems associated with implementation of a fully flow dependent algorithm such as the extended and ensemble Kalman filters.

The utility of the new algorithm was demonstrated in chapter 7. The scheme was implemented in three simple dynamical system models with a range of characteristics. The results were extremely positive; we were able to recover the true parameter values to a good level of accuracy, even when observations were noisy. This had a positive impact on the predictive skill of the forecast model. In the experiments with noisy observations we found averaging the parameters over a moving time window to be very effective at smoothing the parameter updates and therefore yielding more accurate estimates (provided the scheme was given an initial lead-in period in which to settle).

The potential applicability of the hybrid technique to morphodynamic modelling was investigated in chapter 8. Here we tested the methodology in an idealised two parameter 1D nonlinear sediment transport model based on the sediment conservation equation introduced in chapter 4. This was a particularly valuable exercise as it required consideration of issues not encountered with the previous models. The formulation of this model is such that the model parameters exhibit strong interdependence. This means that both parameters have a similar effect on the model and thus causes difficulties relating to identifiability and non-uniqueness of solutions as discussed in chapter 3. This highlighted the importance of consideration of the relationship between individual parameters as described by the parameter cross correlations, and raised the issue of how to specify the parameter error covariance matrix \mathbf{B}_{pp} . For the simple model experiments in chapter 7 the parameter cross covariances were assumed to be zero and this assumption proved sufficient. For the 1D sediment transport model we found that, to obtain

parameter estimates that matched the true values, non-zero parameter cross covariances were needed and crucially that these must reflect the direction of the parameter correlation. It was noted that in some cases it may be appropriate to re-parameterise or restructure the model or to consider transforming the parameters to a set of uncorrelated variables.

A further issue for consideration was how to approximate the Jacobian matrix \mathbf{N}_k required for the computation of the state-parameter cross covariance matrix \mathbf{B}_{zp} . Unlike for the models in chapter 7, an explicit computational form for \mathbf{N}_k was not available. We therefore proposed using a local finite difference approximation; this was found to work well and did not significantly add to the overall cost of the scheme.

As we would expect, there are bounds on the success of the approach. If observational data are too infrequent in space and time or too noisy, or if the error in the initial state and parameter background estimates is large then we cannot expect the scheme to yield reliable results. For the linear advection and nonlinear sediment transport models the positioning of the observations in relation to key features also had an effect. Generally, we saw a deterioration in the parameter estimates and model predictions as the quality and frequency of the observation decreased. Results varied depending on the particular characteristics of each individual model, but the confines on the scheme were not found to be unreasonable for any of the cases considered here.

This lead us back to the concept of observability, that is, whether the observations contain sufficient information to determine the parameters uniquely. Also linked to this is parameter sensitivity. We found that we were less easily able to identify parameters to which a model is relatively insensitive, as was the case under certain conditions for the parameter n in the 1D sediment transport model in chapter 8 and for certain values of the damping parameter d in the oscillating system in chapter 7. This is not surprising as we cannot expect to be able to correct parameter errors that cause errors in the model solution that are on smaller scales than those observed. Further work is needed to be able to reach more formal conclusions; parameter observability and conditions for uniqueness need to be considered in more depth in order to establish how they relate to our hybrid scheme.

Overall, the application of sequential data assimilation to the problem of model pa-

parameter estimation has proved extremely successful. The state augmentation technique facilitates efficient joint state-parameter estimation, thereby saving on calibration time, making better use of the available data and delivering more accurate model forecasts. The novel hybrid algorithm developed in this work offers an effective and versatile approach to approximating the state-parameter cross covariances demanded by the augmented system. We expect that there is great potential for successful application of this new methodology to larger, more complex models across a range of environmental sciences.

9.2 Further work

In this work we have used simple, low order models; this allowed us to concentrate on developing, testing and evaluating ideas rather than dealing with modelling complexities. The experiments in chapters 7 and 8 provided a valuable insight into how our proposed new hybrid method is likely to perform in a range of dynamical systems. Ultimately, we hope to use the experience gained from work with these simple models to help guide the application of data assimilation based state and parameter estimation in operational forecasting systems. A natural next step is to develop and test our new hybrid technique in the context of a large, more realistic system.

In our experiments we used synthetic observations that were direct, evenly spaced and assimilated at regular time intervals. In practical applications, systems are typically modelled as at least 2D and run on much larger domains and over much longer timescales. In the case of morphodynamic modelling, the state vector may include additional variables such as waves and currents. The number of uncertain parameters will depend on the choice and complexity of the model. Observational data are also generally non-uniformly distributed in space and less frequent in time.

Working with a more realistic model would provide the opportunity to gain a greater insight into some of these issues and to assess the feasibility of our augmented hybrid approach in the context of operational scale modelling. A more complex model will need a more sophisticated state background covariance model and the assimilation of real observations will require the consideration of issues such as nonlinear observation

operators, representativity and instrument errors. The type and frequency of availability of observational data is also an important factor. We found that the quality of the state and parameter estimates varied with the spatial and temporal frequency of the observations. Further investigation is needed to establish whether the observation space and time frequencies available on an operational level are sufficient for the approach to be effective. One way of doing this would be to perform some simple observing system simulation experiments (OSSEs) [1], [14]. This would help us to understand how the analysis accuracy changes with different observation sets and help to identifying the optimal observational network design. Alternatively, this question could be addressed using one of the many adaptive observation or observation targetting strategies employed in the meteorological community e.g. [8], [52].

A further interesting study would be to compare our new parameter estimation method against traditional morphodynamic model calibration techniques or against other automated algorithms as was done in [101] for a simple biogeochemical model. This would enable us to evaluate whether this new approach offers a cost effective alternative to methods currently in use.

Ideally we would like to be able to establish some formal results in respect of the convergence and stability of this new method. For this we would need to draw on ideas and methods from control theory. Since most of the basic theory is based on the assumption that the forecast model and observation operators are linear, these issues would first need to be addressed in terms of our simple linear advection model. We would then hope to be able to extend the results to a linearisation of the nonlinear model.

Although the long-term goal is to implement a parameter estimation scheme in a full morphodynamic assimilation-forecast system applied to some real coastal study sites, our new hybrid method could in principle be employed in a variety of environmental systems. The techniques explored in this thesis are also relevant to other data assimilation applications. Parameter estimation can be viewed as a simplification of the more complex problem of model error estimation [16]. A particular advantage of the state augmentation approach is that it allows the errors to take different forms. For different types of error, different error evolution models will be appropriate. It would be interesting to investigate whether the current parameter estimation technique can be extended

to account for different types of error, for example to allow for model errors that evolve with time.

Appendix A

Tangent Linear Model (TLM)

A.1 Definition

If \mathbf{f} is a nonlinear model defined as

$$\mathbf{z}_{k+1} = \mathbf{f}(\mathbf{z}_k),$$

then the *tangent linear model* of \mathbf{f} at \mathbf{z}_k , called \mathbf{F} is

$$\delta \mathbf{z}_{k+1} = \mathbf{F}_k \delta \mathbf{z}_k = \frac{\partial \mathbf{f}(\mathbf{z}_k)}{\partial \mathbf{z}} \delta \mathbf{z}_k.$$

A.2 Tangent Linear of the augmented system model

Starting from an initial state $\widehat{\mathbf{w}}_k$ at time t_k we generate a reference state at t_{k+1} using the model equation (3.3)

$$\widehat{\mathbf{w}}_{k+1} = \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k). \quad (\text{A.1})$$

We define a perturbation to this state as

$$\delta \mathbf{w}_{k+1} = \mathbf{w}_{k+1} - \widehat{\mathbf{w}}_{k+1}. \quad (\text{A.2})$$

This perturbation then satisfies

$$\delta \mathbf{w}_{k+1} = \tilde{\mathbf{f}}(\mathbf{w}_k) - \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k). \quad (\text{A.3})$$

Assuming $\delta \mathbf{w}_{k+1}$ is small, we can expand (A.3) in a Taylor series about $\widehat{\mathbf{w}}_{k+1}$. To first order we have

$$\begin{aligned}
\delta \mathbf{w}_{k+1} &= \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k + \delta \mathbf{w}_k) - \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k) \\
&= \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k) + \mathbf{F}_k \delta \mathbf{w}_k + \dots - \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k) \\
&\approx \mathbf{F}_k \delta \mathbf{w}_k,
\end{aligned} \tag{A.4}$$

where

$$\mathbf{F}_k = \frac{\partial \tilde{\mathbf{f}}(\widehat{\mathbf{w}}_k)}{\partial \mathbf{w}}, \tag{A.5}$$

is the Jacobian of the forecast model with respect to \mathbf{w} evaluated at $\widehat{\mathbf{w}}_k$.

Thus we can approximate

$$\tilde{\mathbf{f}}_k(\mathbf{w}_k) - \tilde{\mathbf{f}}_k(\widehat{\mathbf{w}}_k) \approx \mathbf{F}_k(\mathbf{w}_k - \widehat{\mathbf{w}}_k) \tag{A.6}$$

Note that this approximation is only valid if the perturbations to the model state are small, i.e. $\|\mathbf{w} - \widehat{\mathbf{w}}_k\|_2 < \epsilon$ for some small $\epsilon > 0$.

Appendix B

Block Matrix Inversion

If we partition the augmented background error covariance matrix \mathbf{B} into block form

$$\mathbf{B} = \begin{pmatrix} \mathbf{B}_{zz} & \mathbf{B}_{zp} \\ (\mathbf{B}_{zp})^T & \mathbf{B}_{pp} \end{pmatrix}, \quad (\text{B.1})$$

we can invert \mathbf{B} blockwise using the analytic inversion formula [80]:

$$\begin{aligned} \mathbf{B}^{-1} &= \begin{pmatrix} \mathbf{B}_{zz} & \mathbf{B}_{zp} \\ (\mathbf{B}_{zp})^T & \mathbf{B}_{pp} \end{pmatrix}^{-1} \\ &= \begin{pmatrix} \mathbf{B}_{zz}^{-1} + \mathbf{B}_{zz}^{-1} \mathbf{B}_{zp} \mathbf{S}_z^{-1} (\mathbf{B}_{zp})^T \mathbf{B}_{zz}^{-1} & -\mathbf{B}_{zz}^{-1} \mathbf{B}_{zp} \mathbf{S}_z^{-1} \\ -\mathbf{S}_z^{-1} (\mathbf{B}_{zp})^T \mathbf{B}_{zz}^{-1} & \mathbf{S}_z^{-1} \end{pmatrix}, \quad (\text{B.2}) \end{aligned}$$

where

$$\mathbf{S}_z = \mathbf{B}_{pp} - (\mathbf{B}_{zp})^T \mathbf{B}_{zz}^{-1} \mathbf{B}_{zp}$$

is the called the Schur complement of \mathbf{B}_{zz} [80].

Alternatively this can be written as

$$\mathbf{B}^{-1} = \begin{pmatrix} \mathbf{S}_p^{-1} & -\mathbf{S}_p^{-1} \mathbf{B}_{zp} \mathbf{B}_{pp}^{-1} \\ -\mathbf{B}_{pp}^{-1} (\mathbf{B}_{zp})^T \mathbf{S}_p^{-1} & \mathbf{B}_{pp}^{-1} + \mathbf{B}_{pp}^{-1} (\mathbf{B}_{zp})^T \mathbf{S}_p^{-1} \mathbf{B}_{zp} \mathbf{B}_{pp}^{-1} \end{pmatrix}. \quad (\text{B.3})$$

where

$$\mathbf{S}_p = \mathbf{B}_{zz} - \mathbf{B}_{zp} \mathbf{B}_{pp}^{-1} (\mathbf{B}_{zp})^T$$

is the Schur complement of \mathbf{B}_{pp} .

Appendix C

The Upwind scheme for the linear advection equation

The upwind scheme for the linear advection equation (7.1) with $c > 0$ and periodic boundary conditions is given by

$$z_j^{k+1} = (1 - c\mu)z_j^k + c\mu z_{j-1}^k, \quad j = 1, 2, \dots, m \quad k = 0, 1, \dots, T \quad (\text{C.1})$$

with

$$z_{k,0} = z_{k,J} \quad (\text{C.2})$$

where $z_{k,j} \approx z(x_j, t_k)$ with $x_j = j\Delta x$, $t_k = k\Delta t$ and $\mu = \frac{\Delta t}{\Delta x}$.

The scheme is first order accurate and stable provided that the CFL condition $c\mu \leq 1$ is satisfied [18].

C.1 Model validation

The model is validated against the analytical solution (5.3). The quality of the numerical solution varies with Δx and Δt . Figure C.1 (top) compares the exact (solid red line) and upwind numerical (dashed blue line) solutions for $c = 0.5$ using Gaussian initial data (7.17) on the domain $x \in [0, 3]$ at times $t = 1, 2$ and 3 with grid spacing $\Delta x = 0.01$ and time step $\Delta t = 0.01$. The upwind scheme suffers from numerical dissipation; this causes

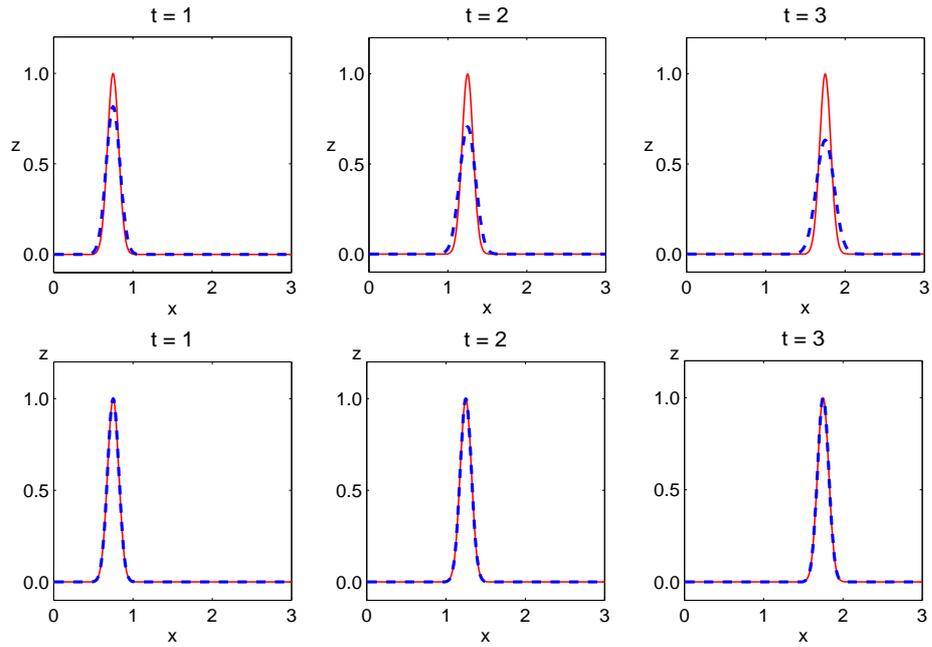


Figure C.1: Comparison of analytic and numerical solutions to the linear advection equation (5.3) for $c = 0.5$ with Gaussian initial data at times $t = 1, 2$ & 3 . **Top:** $c\mu = 0.5$, **Bottom:** $c\mu = 1.0$. The solid red line represents the analytic solution, the dashed blue line is the solution computed using the Upwind scheme.

a smearing of the solution over time so that our initial data profile gradually reduces in height and increases in width. However, the scheme is non dispersive which means that the numerical solution travels at the correct speed. The accuracy of the solution can be improved by refining the model grid. With $c\mu = 1$ the discretisation (C.1) yields the true solution of the linear advection equation. The rate of dissipation can be decreased by increasing the value of $c\mu$ towards the upper boundary of the stability region. The bottom three panels of figure C.1 compare the exact and upwind numerical solutions with grid spacing $\Delta x = 0.01$ and time step $\Delta t = 0.02$, giving $c\mu = 1$. Here the two solutions are indistinguishable.

Appendix D

Jacobian matrix calculations

The matrix $\mathbf{N}_k \in \mathbb{R}^{m \times q}$ is defined as the Jacobian or derivative of the state forecast model with respect to the parameter vector evaluated at the current analysis state \mathbf{x}_k^a , \mathbf{p}_k^a

$$\mathbf{N}_k = \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{p})}{\partial \mathbf{p}} \right|_{\mathbf{x}_k^a, \mathbf{p}_k^a}, \quad (\text{D.1})$$

where $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{p} \in \mathbb{R}^q$.

D.1 Nonlinear oscillating system

Defining the state and parameter vectors

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \text{and} \quad \mathbf{p} = \begin{pmatrix} d \\ m \end{pmatrix}, \quad (\text{D.2})$$

where $y = dx/dt$, we discretise the oscillating system (section 7.2) as follows

$$x_{k+1} = \left(\Delta t - d \frac{\Delta t^2}{2} \right) y_k + \left(1 - m \frac{\Delta t^2}{2} - \frac{\Delta t^2}{2} x_k^2 \right) x_k, \quad (\text{D.3})$$

$$y_{k+1} = \left(1 - d \Delta t - m \frac{\Delta t^2}{2} + d^2 \frac{\Delta t^2}{2} \right) y_k + \left(-m \Delta t + dm \frac{\Delta t^2}{2} + \left(d \frac{\Delta t^2}{2} - \frac{\Delta t}{2} \right) x_k^2 \right) x_k - \frac{\Delta t}{2} (x_k + \Delta t y_k)^3. \quad (\text{D.4})$$

For this system, \mathbf{N}_k is the 2×2 matrix

$$\mathbf{N}_k = \left(\begin{array}{cc} \frac{\partial x_{k+1}}{\partial d} & \frac{\partial x_{k+1}}{\partial m} \\ \frac{\partial y_{k+1}}{\partial d} & \frac{\partial y_{k+1}}{\partial m} \end{array} \right) \Big|_{d_k, m_k} \quad (\text{D.5})$$

The elements of the matrix (D.5) can be calculated directly from (D.3) and (D.4) as

$$\frac{\partial x_{k+1}}{\partial d} = -\frac{\Delta t^2}{2} y_k \quad (\text{D.6})$$

$$\frac{\partial x_{k+1}}{\partial m} = -\frac{\Delta t^2}{2} x_k \quad (\text{D.7})$$

$$\frac{\partial y_{k+1}}{\partial d} = (d\Delta t^2 - \Delta t) y_k + (x_k^2 + m) \frac{\Delta t^2}{2} x_k \quad (\text{D.8})$$

$$\frac{\partial y_{k+1}}{\partial m} = -\frac{\Delta t^2}{2} y_k + \left(-\Delta t + d \frac{\Delta t^2}{2} \right) x_k. \quad (\text{D.9})$$

The matrix (D.5) is recomputed at each new assimilation time by evaluating equations (D.6) - (D.9) at the current parameter estimates d_k, m_k .

D.2 Lorenz equations

For the Lorenz system (section 7.3) we have the state and parameter vectors

$$\mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \quad \text{and} \quad \mathbf{p} = \begin{pmatrix} \sigma \\ \rho \\ \beta \end{pmatrix}. \quad (\text{D.10})$$

The discrete equations for the evolution of the model state variables x , y and z are

$$x_{k+1} = x_k + \sigma \frac{\Delta t}{2} \left[2(y_k - x_k) + \Delta t(\rho x_k - y_k - x_k z_k) - \sigma \Delta t(y_k - x_k) \right], \quad (\text{D.11})$$

$$\begin{aligned} y_{k+1} = & y_k + \frac{\Delta t}{2} \left[\rho x_k - y_k - x_k z_k + \rho(x_k + \sigma \Delta t(y_k - x_k)) - y_k - \Delta t(\rho x_k - y_k - x_k z_k) \right. \\ & \left. - (x_k + \sigma \Delta t(y_k - x_k))(z_k + \Delta t(x_k y_k - \beta z_k)) \right] \end{aligned} \quad (\text{D.12})$$

$$\begin{aligned} z_{k+1} = & z_k + \frac{\Delta t}{2} \left[x_k y_k - \beta z_k + (x_k + \Delta t \sigma(y_k - x_k))(y_k + \Delta t(\rho x_k - y_k - x_k z_k)) \right. \\ & \left. - \beta(z_k + \Delta t(x_k y_k - \beta z_k)) \right]. \end{aligned} \quad (\text{D.13})$$

The Jacobian matrix \mathbf{N}_k is given by

$$\mathbf{N}_k = \begin{pmatrix} \frac{\partial x_{k+1}}{\partial \sigma} & \frac{\partial x_{k+1}}{\partial \rho} & \frac{\partial x_{k+1}}{\partial \beta} \\ \frac{\partial y_{k+1}}{\partial \sigma} & \frac{\partial y_{k+1}}{\partial \rho} & \frac{\partial y_{k+1}}{\partial \beta} \\ \frac{\partial z_{k+1}}{\partial \sigma} & \frac{\partial z_{k+1}}{\partial \rho} & \frac{\partial z_{k+1}}{\partial \beta} \end{pmatrix} \bigg|_{\sigma_k, \rho_k, \beta_k}. \quad (\text{D.14})$$

The elements of (D.14) are computed by differentiating the discrete equations (D.11), (D.12) and (D.13) with respect to each of the parameters σ , ρ and β . This yields the following

$$\begin{aligned} \frac{\partial x_{k+1}}{\partial \sigma} &= \frac{\Delta t}{2} \left[2(y_k - x_k) + \Delta t(\rho x_k - y_k - x_k z_k) - 2\sigma \Delta t(y_k - x_k) \right], \\ &= \Delta t(1 - \sigma \Delta t)(y_k - x_k) + \frac{\Delta t^2}{2}(\rho x_k - y_k - x_k z_k) \end{aligned} \quad (\text{D.15})$$

$$\frac{\partial x_{k+1}}{\partial \rho} = \sigma \frac{\Delta t^2}{2} x_k \quad (\text{D.16})$$

$$\frac{\partial x_{k+1}}{\partial \beta} = 0 \quad (\text{D.17})$$

$$\begin{aligned}
\frac{\partial y_{k+1}}{\partial \sigma} &= \frac{\Delta t}{2} \left[\rho \Delta t (y_k - x_k) - \Delta t (y_k - x_k) (z_k + \Delta t (x_k y_k - \beta z_k)) \right], \\
&= \frac{\Delta t^2}{2} (y_k - x_k) \left[\rho - z_k - \Delta t (x_k y_k - \beta z_k) \right]
\end{aligned} \tag{D.18}$$

$$\begin{aligned}
\frac{\partial y_{k+1}}{\partial \rho} &= \frac{\Delta t}{2} \left[x_k + x_k + \sigma \Delta t (y_k - x_k) - \Delta t x_k \right], \\
&= \frac{\Delta t}{2} \left[(2 - \Delta t) x_k + \sigma \Delta t (y_k - x_k) \right], \\
&= \left(\Delta t - \frac{\Delta t^2}{2} \right) x_k - \sigma \frac{\Delta t^2}{2} (y_k - x_k)
\end{aligned} \tag{D.19}$$

$$\begin{aligned}
\frac{\partial y_{k+1}}{\partial \beta} &= \frac{\Delta t}{2} \left[x_k \Delta t z_k + \sigma \Delta t (y_k - x_k) \Delta t z_k \right], \\
&= \frac{\Delta t^2}{2} \left[x_k + \sigma \Delta t (y_k - x_k) \right] z_k
\end{aligned} \tag{D.20}$$

$$\begin{aligned}
\frac{\partial z_{k+1}}{\partial \sigma} &= \frac{\Delta t}{2} \left[\Delta t (y_k - x_k) (y_k + \Delta t (\rho x_k - y_k - x_k z_k)) \right], \\
&= \frac{\Delta t^2}{2} \left[(y_k - x_k) y_k + \Delta t (y_k - x_k) (\rho x_k - y_k - x_k z_k) \right]
\end{aligned} \tag{D.21}$$

$$\begin{aligned}
\frac{\partial z_{k+1}}{\partial \rho} &= \frac{\Delta t}{2} \left[(x_k + \Delta t \sigma (y_k - x_k)) \Delta t x_k \right], \\
&= \frac{\Delta t^2}{2} \left[x_k + \sigma \Delta t (y_k - x_k) \right] x_k
\end{aligned} \tag{D.22}$$

$$\begin{aligned}
\frac{\partial z_{k+1}}{\partial \beta} &= \frac{\Delta t}{2} \left[-z_k - \Delta t x_k y_k + 2\beta \Delta t z_k \right], \\
&= \frac{\Delta t^2}{2} x_k y_k + \frac{\Delta t}{2} (2\beta \Delta t - 1) z_k.
\end{aligned} \tag{D.23}$$

The Jacobian matrix at time t_k is formed by evaluating (D.15) - (D.23) at the current estimated parameter values $\sigma_k, \rho_k, \beta_k$.

Appendix E

SLCN scheme validation

E.1 The semi-Lagrangian scheme

The semi-Lagrangian component of the combined SLCN algorithm 8.1 was validated using the inviscid Burgers' equation

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0, \quad (\text{E.1})$$

on the domain $x \in [-1, 1]$, with initial data

$$u(x, 0) = \begin{cases} 0 & x \leq -1 \\ 1 - x^2 & -1 < x < 1 \\ 0 & x \geq 1 \end{cases} . \quad (\text{E.2})$$

An analytic solution for $u(x, t)$ can be derived using the method of characteristics as described in [88]. The solution for $t > 0$ is given by

$$u(x, t) = u(x_0, 0) = \begin{cases} 0 & x_0 \leq -1 \\ 1 - x_0^2 & -1 < x_0 < 1 \\ 0 & x_0 \geq 1 \end{cases} \quad (\text{E.3})$$

where

$$x_0 = \frac{1 - \sqrt{1 - 4t(x - t)}}{2t}, \quad -1 < x < 1. \quad (\text{E.4})$$

Figure E.1 compares the solution computed using the semi-Lagrangian method with grid spacing $\Delta x = 0.01$ and time step $\Delta t = 0.05$ against the exact solution at times $t = 0$ to $t = 0.5$. As illustrated, the model is in close agreement with the analytical solution.

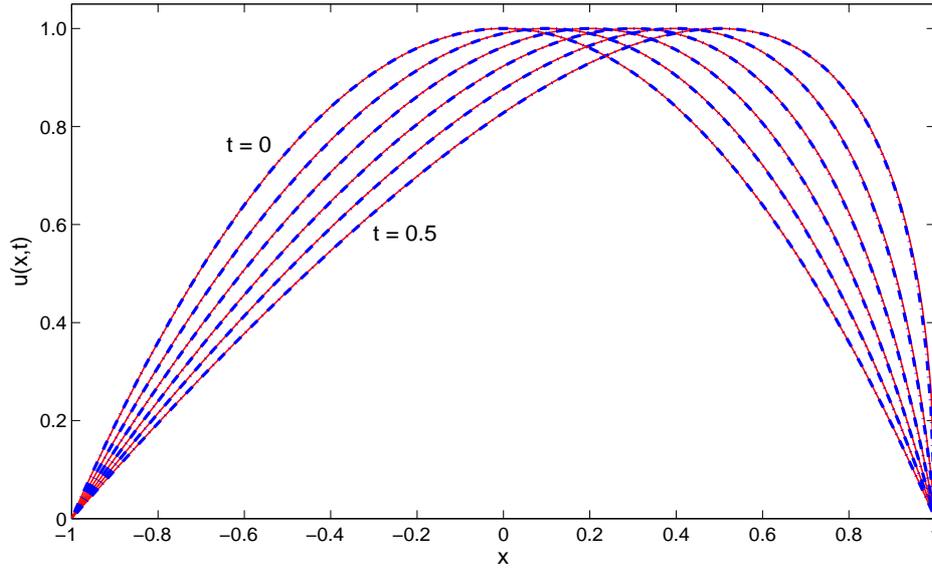


Figure E.1: Comparison of analytic and numerical solutions to the inviscid Burgers' equation (E.1) at times $t = 0, 0.1, 0.2, 0.3, 0.4$ & 0.5 . The solid red line represents the analytic solution, the dot-dash blue line is the solution computed using the semi-Lagrangian scheme with grid spacing $\Delta x = 0.01$ and time step $\Delta t = 0.05$.

E.2 The Crank-Nicolson scheme

The Crank Nicolson component was validated using the heat equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} \quad (\text{E.5})$$

on the domain $x \in [0, \pi]$, subject to the initial data

$$u(x, 0) = \sin x, \quad (\text{E.6})$$

and boundary conditions

$$u(0, t) = u(\pi, t) = 0 \quad t > 0. \quad (\text{E.7})$$

The exact solution for $t > 0$ is given by [11]

$$u(x, t) = e^{-t} \sin x. \quad (\text{E.8})$$

Figure E.2 shows the solution computed using the Crank Nicolson method with grid spacing $\Delta x = \pi/125$ and time step $\Delta t = 0.05$ plotted against the exact solution (E.8) at times $t = 0$ to 0.5 . Again, the numerical solution is extremely accurate.

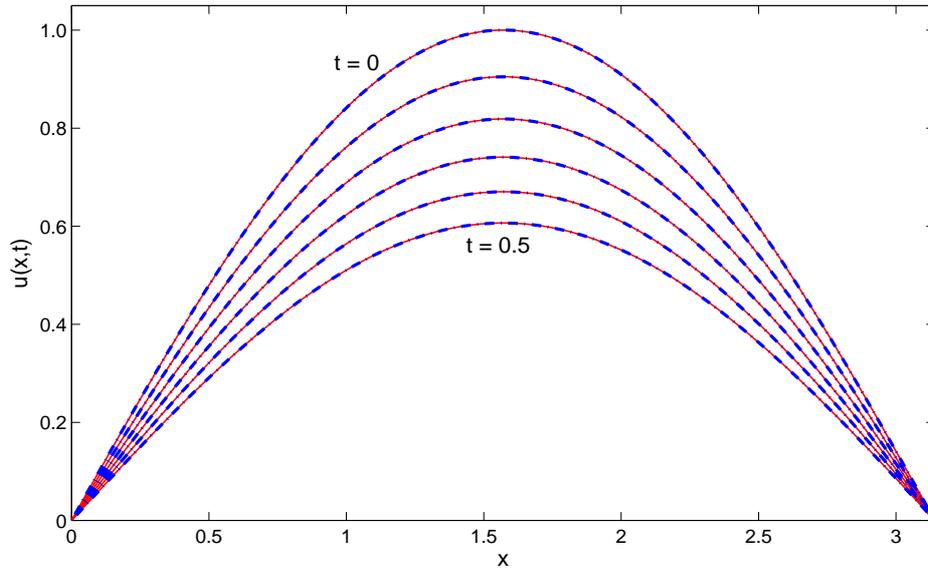


Figure E.2: Comparison of analytic and numerical solutions to the heat equation (E.5) at times $t = 0, 0.1, 0.2, 0.3, 0.4$ & 0.5 . The solid red line represents the analytic solution, the dot-dash blue line is the solution computed using the Crank Nicolson scheme with grid spacing $\Delta x = \pi/125$ and time step $\Delta t = 0.05$.

E.3 Combined semi-Lagrangian Crank-Nicolson scheme

To validate the combined SLCN algorithm we consider the viscous Burgers' equation

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \frac{\partial^2 u}{\partial x^2}, \quad t > 0. \quad (\text{E.9})$$

We use the solution given by [7]

$$u(x, t) = \frac{2 \sin x}{\cos x + e^t} \quad t > 0 \quad (\text{E.10})$$

on $x \in [0, \pi]$, $t \in [0.1, 0.5]$ with boundary conditions

$$u(0, t) = u(\pi, t) = 0. \quad (\text{E.11})$$

The exact and approximate numerical solutions are shown in figure E.3 at times $t = 0.1, 0.2, 0.3, 0.4$ & 0.5 . In this example the model was run with grid spacing $\Delta x = \pi/125$ and time step $\Delta t = 0.05$. The errors for the combined algorithm are marginally greater than in the individual test cases but we are nonetheless satisfied that the scheme provides a suitable method for discretising the advection diffusion model (8.9).

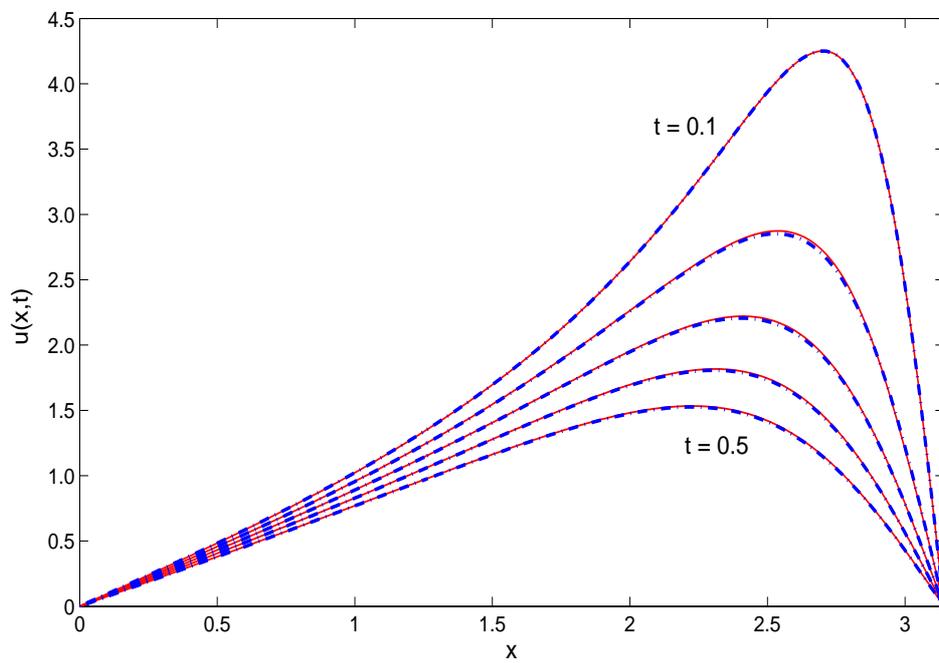


Figure E.3: Comparison of numerical and analytical solutions to the viscous Burgers' equation, at times $t = 0.1, 0.2, 0.3, 0.4$ & 0.5 , on $0 < x < \pi$. The solid red line represents the analytic solution, the dot-dash blue line is the solution computed using the hybrid SLCN scheme with time step $\Delta t = 0.05$ and model grid spacing $\Delta x = \pi/125$.

Bibliography

- [1] C. Arnold and C. Dey. Observation system simulation experiments: Past, present, and future. *Bulletin of the American Meteorological Society*, 67:687–695, 1986.
- [2] R. N. Bannister. A review of forecast error covariance statistics in atmospheric variational data assimilation. I: Characteristics and measurements of forecast error covariances. *Quarterly Journal of Royal Meteorological Society*, 134:1955–1970, 2008.
- [3] R. N. Bannister. A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics. *Quarterly Journal of Royal Meteorological Society*, 134:1971–1996, 2008.
- [4] S. Barnett and R. G. Cameron. *Introduction to Mathematical Control Theory*. Oxford Applied Mathematics and Computing Science Series. Oxford Clarendon Press, second edition, 1990.
- [5] M. J. Bell, M. J. Martin, and N. K. Nichols. Assimilation of data into an ocean model with systematic errors near the equator. *Quarterly Journal of the Royal Meteorological Society*, 130:873–894, 2004.
- [6] P. S. Bell, J. J. Williams, S. Clark, B. D. Morris, and A. Vila-Concejo. Nested radar systems for remote coastal observations. *Journal of Coastal Research*, Special Issue 39 (Proceedings of the ICS 2004), 2005.
- [7] E. R. Benton and G. W. Platzman. A table of solutions of the one-dimensional Burgers' equation. *Quarterly of Applied Mathematics*, 30:195–212, 1972.
- [8] C. H. Bishop, B. J. Etherton, and S. J. Majumbar. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Monthly Weather Review*, 129:420–436, 1991.

- [9] F. Bouttier and P. Courtier. Data assimilation concepts and methods. Meteorological Training Course Lecture Series. ECMWF, 2002.
- [10] J. D. Brown, T. Spencer, and I. Moeller. Modeling storm surge flooding of an urban area with particular reference to modeling uncertainties: A case study of Canvey Island, United Kingdom. *Water Resources Research*, 43(W06402), 2007.
- [11] R. L. Burden and D. J. Faires. *Numerical Analysis*. Brooks-Cole Publishing Company, 6th edition, 1997.
- [12] S. E. Cohn and D. P. Dee. Observability of discretized partial differential equations. *Siam Journal on Numerical Analysis*, 25(3):586–617, 1988.
- [13] P. Courtier, E. Andersson, W. Heckley, J. Pailleux, D. Vasiljevic, M. Hamrud, A. Hollingsworth, F. Rabier, and M. Fisher. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Quarterly Journal of the Royal Meteorological Society*, 124:1783–1807, 1998.
- [14] R. Daley. *Atmospheric Data Analysis*. Cambridge University Press, 1991.
- [15] H. J. de Vriend, M. Capobianco, T. Chesher, H. E. de Swart, B. Latteux, and M. J. F. Stive. Approaches to long-term modelling of coastal morphology: a review. *Coastal Engineering*, 21:225–269, 1993.
- [16] D. P. Dee. Bias and data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 131:3323–2243, 2005.
- [17] D. P. Dee and A. M. Da Silva. Data assimilation in the presence of forecast bias. *Quarterly Journal of the Royal Meteorological Society*, 124:269–295, 1998.
- [18] D. R. Durran. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Texts in Applied Mathematics 32. Springer, 1999.
- [19] R. M. Errico. What is an adjoint model? *Bulletin of the American Meteorological Society*, 78(11):2577–2591, 1997.
- [20] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, 99(C5):10143–10162, 1994.

- [21] G. Evensen. Advanced data assimilation for strongly nonlinear dynamics. *Monthly Weather Review*, 125:1342–1354, 1997.
- [22] G. Evensen, D. P. Dee, and J. Schröter. Parameter estimation in dynamical models. In E. Chassignet and J. Verron, editors, *Ocean Modeling and Parameterization*, pages 373–398. Kluwer Academic Publishers, 1998.
- [23] M. Fisher. Development of a simplified Kalman filter. Technical Memorandum No. 260, ECMWF Research Department, 1998.
- [24] M. Fisher. Background error covariance modelling. In *Seminar on recent developments in data assimilation for atmosphere and ocean*, pages 49–64. ECMWF, 2003.
- [25] P. E. Frandsen, K. Jonasson, H. B. Nielsen, and O. Tingle. Unconstrained optimization. IMM lecture note, Technical University of Denmark, 1999. Available at <http://www2.imm.dtu.dk/documents/ftp/publlec.html> (last accessed June 2010).
- [26] A. Gelb. *Applied Optimal Estimation*. M.I.T Press, 1974.
- [27] M. Ghil and P. Malanotte-Rizzoli. Data assimilation in meteorology and oceanography. *Advances in Geophysics*, 33:141–266, 1991.
- [28] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press, 1981.
- [29] L. Gorman, A. Morang, and R. Larson. Monitoring the coastal environment; Part IV: Mapping, shoreline changes, and bathymetric analysis. *Journal of Coastal Research*, 14(1):61–92, 1998.
- [30] A. J. Grass. Sediment transport by waves and currents. Report No: FL29, SERC London Centre for Marine Technology, 1981.
- [31] A. Griewank and G. F. Corliss. *Automatic Differentiation of Algorithms*. SIAM, Philadelphia, 1991.
- [32] A. K. Griffith. *Data Assimilation for Numerical Weather Prediction Using Control Theory*. PhD thesis, University of Reading, 1997.

- [33] A. K. Griffith and N. K. Nichols. Accounting for model error in data assimilation using adjoint methods. In M. Berz, C. Bischof, G. Corliss, and A. Griewank, editors, *Computational Differentiation: Techniques, Applications and Tools*, pages 195–204. SIAM Philadelphia, 1996.
- [34] A. K. Griffith and N. K. Nichols. Adjoint techniques in data assimilation for treating systematic model error. *Journal of Flow, Turbulence and Combustion*, 65:469–488, 2000.
- [35] J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Applied Mathematical Sciences 42. Springer-Verlag, 1986.
- [36] B. Gutlieus. Engineering applications of airborne scanning lasers: reports from the field. *Photogrammetric Engineering and Remote Sensing*, 64:246–253, 1998.
- [37] A. W. Hesselink, G. S. Stelling, J. C. Kwadijk, and H. Middelkoop. Inundation of a Dutch river polder, sensitivity analysis of a physically based inundation model using historic data. *Water Resources Research*, 39(9):1234, 2003.
- [38] D. C. Hill, S. E. Jones, and D. Prandle. Derivation of sediment resuspension rates from acoustic backscatter time-series in tidal waters. *Continental Shelf Research*, 23:19–40, 2003.
- [39] R. A. Holman and J. Stanley. The history and technical capabilities of Argus. *Coastal Engineering*, 54:477–491, 2007.
- [40] P. L. Houtekamer and H. L. Mitchell. Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, 126:796–811, 1998.
- [41] P. L. Houtekamer and H. L. Mitchell. Ensemble Kalman filtering. *Quarterly Journal of the Royal Meteorological Society*, 131:3269–3289, 2005.
- [42] J. Hudson. *Numerical Techniques for Morphodynamic Modelling*. PhD thesis, University of Reading, 2001.
- [43] J. Hudson, J. Damgaard, N. Dodd, T. Chesher, and A. Cooper. Numerical approaches for 1D morphodynamic modelling. *Coastal Engineering*, 52:691–707, 2005.

- [44] K. P. Ide, P. Courtier, M. Ghil, and A. C. Lorenc. Unified notation for data assimilation: Operational, sequential and variational. *Journal of the Meteorological Society of Japan*, 75(1B):181–189, 1997.
- [45] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, 1970.
- [46] C. Johnson, B. J. Hoskins, N. K. Nichols, and S. P. Ballard. A singular vector perspective of 4D-Var: The spatial structure and evolution of baroclinic weather systems. *Monthly Weather Review*, 134(11):3436–3454, 2006.
- [47] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering (Series D)*, 82:35–45, 1960.
- [48] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME - Journal of Basic Engineering (Series D)*, 83:95–108, 1961.
- [49] E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2003.
- [50] M. A. F. Knaapen and S. J. M. H. Hulscher. Use of genetic algorithm to improve prediction of alternate bar dynamics. *Water Resources Research*, 39(9):1231, 2003.
- [51] G. R. Lesser, J. A. Roelvink, J. A. T. M. van Kester, and G. S. Stelling. Development and validation of a three-dimensional morphological model. *Coastal Engineering*, 51:883–915, 2004.
- [52] M. Leutbecher, J. Barkmeijer, T. N. Palmer, and A. J. Thorpe. Potential improvement to forecasts of two severe storms using targeted observations. *Quarterly Journal of the Royal Meteorological Society*, 128:1641–1670, 2002.
- [53] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhäuser-Verlag, 1992.
- [54] J. M. Lewis, S. Lakshmivarahan, and S. K. Dhall. *Dynamic Data Assimilation: A Least Squares Approach*, volume 104 of *Encyclopedia of Mathematics and its applications*. Cambridge University Press, 2006.
- [55] W. Long, J. T. Kirby, and Z. Shao. A numerical scheme for morphological bed level calculations. *Coastal Engineering*, 55:167–180, 2008.

- [56] A. C. Lorenc. A global three-dimensional multivariate statistical interpolation scheme. *Monthly Weather Review*, 109:701–721, 1981.
- [57] A. C. Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112:1177–1194, 1986.
- [58] A. C. Lorenc, S. P. Ballard, R. S. Bell, N. B. Ingleby, P. L. F. Andrews, D. M. Barker, J. R. Bray, A. M. Clayton, T. Dalby, D. Li, T. J. Payne, and F. W. Saunders. The Met Office global three-dimensional variational data assimilation scheme. *Quarterly Journal of the Royal Meteorological Society*, 126:2991–3012, 2000.
- [59] A. C. Lorenc and F. Rawlins. Why does 4D-Var beat 3D-Var? *Quarterly Journal of the Royal Meteorological Society*, 131:3247–3257, 2005.
- [60] E. N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20:130–141, 1963.
- [61] M. Martin and A. Lawless. *Oscillating system with sequential data assimilation scheme*. NERC National Centre for Earth Observation. Available at <http://www.nceo.ac.uk/training.php>. (last accessed June 2010).
- [62] M. J. Martin. *Data Assimilation in Ocean Circulation Models with Systematic Errors*. PhD thesis, University of Reading, 2000.
- [63] M. J. Martin, M. J. Bell, and N. K. Nichols. Estimation of systematic error in an equatorial ocean model using data assimilation. *International Journal for Numerical Methods in Fluids*, 40:435–444, 2002.
- [64] M. J. Martin, N. K. Nichols, and M. J. Bell. Treatment of systematic errors in sequential data assimilation. Technical Note No. 21, Meteorological Office, Ocean Applications Division, 1999.
- [65] D. C. Mason, M. Amin, I. J. Davenport, R. A. Flather, G. J. Robinson, and J. A. Smith. Measurement of recent intertidal sediment transport in Morecambe Bay using the waterline method. *Estuarine, Coastal and Shelf Science*, 49:427–456, 1999.
- [66] D. C. Mason, I. J. Davenport, R. A. Flather, C. Gurney, G. J. Robinson, and J. A. Smith. A sensitivity analysis of the waterline method of constructing a digital

- elevation model for intertidal areas in an ERS SAR scene of eastern England. *Estuarine, Coastal and Shelf Science*, 53:759–778, 2001.
- [67] D. C. Mason, C. Gurney, and M. Kennett. Beach topography mapping - a comparison of techniques. *Journal of Coastal Conservation*, 6:113–124, 2000.
- [68] G. Masselink and M. G. Hughes. *Introduction to Coastal Processes and Geomorphology*. Hodder Arnold, 2003.
- [69] 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.
- [70] H. Moradkhani, S. Sorooshian, H. V. Gupta, and P. R. Houser. Dual state-parameter estimation of hydrological models using ensemble Kalman filter. *Advances in Water Resources*, 28:135–147, 2005.
- [71] K. Morton and D. Mayers. *Numerical solution of partial differential equations*. Cambridge University Press, 2nd edition, 2005.
- [72] I. M. Navon. Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. *Dynamics of Atmosphere and Oceans*, 27:55–79, 1997.
- [73] R. J. Nicholls, P. P. Wong, V. R. Burkett, J. O. Codignotto, J. E. Hay, R. F. McLean, S. Ragoonaden, and C. D. Woodroffe. Chapter 6: Coastal systems and low-lying areas. In *Climate Change 2007: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*, pages 315–356. Cambridge University Press, 2007.
- [74] N. K. Nichols. Data assimilation: Aims and basic concepts. In R. Swinbank, V. Shutyaev, and W. A. Lahoz, editors, *Data Assimilation for the Earth System*, volume 26 of *Nato Science Series IV: Earth & Environmental Sciences*, pages 9–20. Kluwer Academic, 2003.
- [75] N. K. Nichols. Mathematical concepts of data assimilation. In W. A. Lahoz, R. Swinbank, and B. Khatatov, editors, *Data Assimilation: Making Sense of Observations*. Springer, 2009.

- [76] P. Nielsen. *Coastal bottom boundary layers and sediment transport*, volume 4 of *Advanced Series on Ocean Engineering*. World Scientific, 2005.
- [77] J. Nocedal. Theory of algorithms for unconstrained optimization. *Atica Numerica*, 1992.
- [78] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, New York, 1999.
- [79] D. F. Parrish and J. C. Derber. The National Meteorological Center’s spectral statistical-interpolation analysis system. *Monthly Weather Review*, 120:1747–1763, 1992.
- [80] K. B. Petersen and M. S. Pedersen. The Matrix Cookbook. Technical University of Denmark, October 2008. Available at <http://www2.imm.dtu.dk/pubdb/p.php?3274> (last accessed June 2010).
- [81] L. Pinto, B. Andre, and P. F. Fortunato. Sensitivity analysis of non-cohesive sediment transport formulae. *Continental Shelf Research*, 26:1826–1839, 2006.
- [82] C. D. Rodgers. *Inverse Methods for Atmospheric Sounding: Theory and Practice*, volume 2 of *Series on Atmospheric, Oceanic and Planetary Physics*. World Scientific, 2000.
- [83] J. A. Roelvink. Coastal morphodynamic evolution techniques. *Coastal Engineering*, 53:277–287, 2006.
- [84] B. Ruessink. Calibration of nearshore process models - application of a hybrid genetic algorithm. *Journal of Hydroinformatics*, 7:135–149, 2005.
- [85] B. Ruessink. Predictive uncertainty of a nearshore bed evolution model. *Continental Shelf Research*, 25:1053–1069, 2005.
- [86] B. Ruessink. A Bayesian estimation of parameter-induced uncertainty in a nearshore alongshore current model. *Journal of Hydroinformatics*, 7:37–49, 2006.
- [87] T. R. Scott and D. C. Mason. Data assimilation for a coastal area morphodynamic model: Morecambe Bay. *Coastal Engineering*, 54:91–109, 2007.
- [88] P. J. Smith, M. J. Baines, S. L. Dance, N. K. Nichols, and T. R. Scott. Simple models of changing bathymetry with data assimilation. Numerical Analysis Report

- 10/2007, Department of Mathematics, University of Reading, 2007. Available at <http://www.reading.ac.uk/maths/research/>.
- [89] P. J. Smith, M. J. Baines, S. L. Dance, N. K. Nichols, and T. R. Scott. Data assimilation for parameter estimation with application to a simple morphodynamic model. Mathematics Report 2/2008, Department of Mathematics, University of Reading, 2008. Available at <http://www.reading.ac.uk/maths/research/>.
- [90] P. J. Smith, M. J. Baines, S. L. Dance, N. K. Nichols, and T. R. Scott. Variational data assimilation for parameter estimation: application to a simple morphodynamic model. *Ocean Dynamics*, 59(5):697–708, 2009.
- [91] P. J. Smith, S. L. Dance, and N. K. Nichols. Data assimilation for morphodynamic model parameter estimation: a hybrid approach. Mathematics Report 2/2009, Department of Mathematics, University of Reading, 2009. Available at <http://www.reading.ac.uk/maths/research/>.
- [92] P. J. Smith, S. L. Dance, and N. K. Nichols. A hybrid sequential data assimilation scheme for model state and parameter estimation. Mathematics Report 2/2010, Department of Mathematics, University of Reading, 2010. Available at <http://www.reading.ac.uk/maths/research/>.
- [93] S. Sorooshian and V. K. Gupta. Model calibration. In V. J. Singh, editor, *Computer models of watershed hydrology*, chapter 2, pages 23–68. Water Resources Publications, Colorado, 1995.
- [94] R. Soulsby. *Dynamics of marine sands*. Thomas Telford Publications, 1997.
- [95] C. Sparrow. *The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors*. Applied Mathematical Sciences 41. Springer-Verlag, 1982.
- [96] M. Spiegelman and R. F. Katz. A semi-Lagrangian Crank-Nicolson algorithm for the numerical solution of advection-diffusion problems. *Geochemistry Geophysics Geosystems*, 7(4), 2006.
- [97] A. Staniforth and J. Côté. Semi-Lagrangian schemes for atmospheric models - a review. *Monthly Weather Review*, 119(9):2206–2223, 1991.
- [98] J. J. Stoker. *Nonlinear Vibrations in Mechanical and Electrical Systems*. Pure and Applied Mathematics Volume II. Interscience, 1966.

- [99] J. M. T. Thompson and H. B. Stewart. *Nonlinear Dynamics and Chaos: Geometrical Methods for Engineers and Scientists*. John Wiley and Sons Ltd., 1986.
- [100] C. Trudinger, M. Raupach, P. Rayner, and I. Enting. Using the Kalman filter for parameter estimation in biogeochemical models. *Environmetrics*, 19(8):849–870, 2008.
- [101] C. M. Trudinger, M. R. Raupach, P. J. Rayner, J. Kattge, Q. Liu, B. Pak, M. Reichstein, L. Renzullo, A. D. Richardson, S. H. Roxburgh, J. Styles, Y. P. Wang, P. Briggs, D. Barrett, and S. Nikolova. OptIC: An intercomparison of optimization techniques for parameter estimation in terrestrial biogeochemical models. *Journal of Geophysical Research*, 112(G02027), 2007.
- [102] A. van Dongeren, N. Plant, A. Cohen, D. Roelvink, M. C. Haller, and P. Catalán. Beach Wizard: Nearshore bathymetry estimation through assimilation of model computations and remote observations. *Coastal Engineering*, 55:1016–1027, 2008.
- [103] L. C. van Rijn. *Principles of Sediment Transport in Rivers, Estuaries and Coastal Seas*. Aqua Publications, 1993.
- [104] J. A. Vrugt, C. G. H. Diks, H. V. Gupta, W. Bouten, and J. M. Verstraten. Improved treatment of uncertainty in hydrologic modeling: Combining the strengths of global optimization and data assimilation. *Water Resources Research*, 41(1), 2005. W01017.
- [105] G. B. Whitham. *Linear and Nonlinear Waves*. John Wiley & Sons, 1974.
- [106] S. Wiggins. *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Texts in Applied Mathematics 2. Springer-Verlag, 1990.
- [107] J. C. Wüst. Data-driven probabilistic predictions of sand wave bathymetry. In H. S. J. M. H., T. Garlan, and D. Idier, editors, *Marine Sandwave and River Dune Dynamics II*, 2004. Proceedings of International Workshop, University of Twente.
- [108] D. Zupanski and M. Zupanski. Model error estimation employing an ensemble data assimilation approach. *Monthly Weather Review*, 134, 2006.