The University of Reading

A hybrid sequential data assimilation scheme for model state and parameter estimation

P.J. Smith, S.L. Dance and N.K. Nichols

MATHEMATICS REPORT 2/2010

Department of Mathematics
The University of Reading
Whiteknights, PO Box 220
Reading, RG6 6AX

Department of Mathematics

A hybrid sequential data assimilation scheme for model state and parameter estimation

P.J. Smith, S.L. Dance, N.K. Nichols

February 2010

Abstract

We present a new hybrid method for concurrent model state and parameter estimation. The new algorithm uses ideas from three dimensional variational data assimilation (3D Var) and the extended Kalman filter (EKF) together with the technique of state augmentation to estimate uncertain model parameters alongside the model state variables in a sequential assimilation-forecast system. The method is relatively simple to implement and computationally inexpensive to run. We demonstrate its efficacy via a series of identical twin experiments with three simple dynamical system models. The scheme is able to recover the true parameter values to a good level of accuracy, even when observational data are noisy. We expect this new technique to be easily transferable to larger, more complex models.

1 Introduction

A numerical model can never completely describe the complex physical processes underlying the behaviour of a real world dynamical system. State of the art computational 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 our model to accurately predict the true system state. Parameterizations 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. The consequence of this is that model parameters often do not represent directly measurable quantities. Therefore, a key question in model development is how to estimate these parameters a priori. Evensen et al. (1998) gives a useful general introduction the problem of parameter estimation in dynamical models. Generally, parameters are determined theoretically or by adhoc calibration of the model against observations. Various other parameter optimization methods have been developed such as downhill simplex optimization (Hill et al. (2003)), genetic algorithm (Knaapen and Hulscher (2003)), hybrid genetic algorithm (Ruessink (2005a)), classical Bayesian (Wüst (2004)) and Bayesian Generalised Likelihood Uncertainty Estimation (GLUE) (Ruessink (2005b), Ruessink (2006)). An alternative approach is to use data assimilation.

Data assimilation is a sophisticated mathematical technique for combining observational data with model predictions. It is most commonly used 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. State augmentation is a conceptually straightforward technique that enables us to estimate and update uncertain model parameters jointly with the model state variables as part of the assimilation process (Jazwinski (1970)). The approach has previously been used in the context of model error or bias estimation (See e.g. Bell et al. (2004), Dee (2005), Griffith and Nichols (1996), Griffith and Nichols (2000), Martin et al. (2002)).

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. Navon (1997) and Evensen et al. (1998) review the use of the technique in the context of four dimensional variational data assimilation (4D Var). State augmentation has also been applied with the Kalman filter; Martin (2000) uses the method for model bias estimation and Trudinger et al.

(2008) combine the technique with the extended and ensemble Kalman filters for parameter estimation in biogeochemical models. In Zupanski and Zupanski (2006) the approach is combined with the maximum likelihood ensemble filter (MLEF) for both parameter and model bias estimation in the Korteweg-de Vries-Burgers (KdVB) model.

In this work we combine the state augmentation method with a three dimensional variational data assimilation (3D Var) scheme. Under certain statistical assumptions, the 3D Var assimilation method approximates the Bayesian maximum a posteriori likelihood estimate of the state and parameters of the system. A key difficulty in the construction of a data assimilation algorithm is specification of the background error covariances. These are used to describe the error statistics of the model predicted state and determine the weight given to the model in the assimilation. A particular issue highlighted by Smith et al. (2008) and Smith et al. (2009a) is the role of the cross-covariances between the state and parameter errors; for successful parameter estimation, it is crucial that these cross-covariances are given a good a priori specification. Conventional 3D Var assumes that the error covariances are stationary; the structure of the background error covariance matrix is specified at the start of the assimilation and kept fixed throughout. In Smith et al. (2008) it was found that whilst this assumption was sufficient for state estimation, it was insufficient for parameter estimation. In order to yield reliable estimates of the true parameters, a flow dependent representation of the state-parameter cross-covariances is required.

Updating the background error covariance matrix at every time step is computationally very expensive, and impracticable when the system of interest is of high dimension. To overcome this problem we have combined ideas from 3D-Var and the extended Kalman filter to produce a new hybrid assimilation scheme that captures the flow dependent nature of the state-parameter cross covariances without explicitly propagating the full system covariance matrix. As we demonstrate here, the method has proved to be applicable to a range of dynamical system models. An additional example of its application is given in Smith et al. (2009b).

In this paper we give details of the formulation of our new hybrid scheme and demonstrate its efficacy using three simple models: a single parameter 1D linear advection model, a two parameter non-linear damped oscillating system, and a three parameter non-linear chaotic system.

The scheme has been tested by running a series of identical twin experiments using pseudo-observations with a range of spatial and temporal frequencies. The results are positive and confirm that our new scheme can indeed be a useful tool in identifying uncertain model parameters. We are able to recover the true parameter values to a good level of accuracy, even when observations are noisy. We believe that there is potential for successful application of our new methodology to larger, more realistic models with more complex parameterizations.

This paper is organised as follows. In section 2 we explain the state augmentation approach and introduce the augmented system model. In section 3 we give an overview of the 3D Var and Kalman filter algorithms upon which our new hybrid scheme is based. Details of the the formulation of the hybrid method are given in section 4. We introduce our models in section 5 and use the methods described in section 4 to derive estimates for the state-parameter cross covariances in each specific case. The results of our experiments are also presented. Finally, in section 6 we summarise the conclusions from this work.

2 Data assimilation and state augmentation

In this section we lay the foundations for understanding the methods we present in subsequent sections. We start by introducing the general system model and explaining the data assimilation notation and terminology that we will be using throughout this paper.

2.1 The model system equations

We consider the discrete non-linear time invariant dynamical system model

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{p}) \qquad k = 0, 1, \ldots \tag{2.1}$$

The vector $\mathbf{x}_k \in \mathbb{R}^n$ is known as the state vector and represents the model state at time t_k, $\mathbf{p} \in \mathbb{R}^q$ is a vector of q uncertain model parameters, and $\mathbf{f} : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ is a non-linear operator describing the evolution of the state from time t_k to t_{k+1}.

In this paper we use the 'perfect' model assumption; that is, we assume that the dynamical system we are studying can be represented on a discrete grid and that the model (2.1) gives an exact description of the true behaviour of this system on the grid. We also assume that $\mathbf{f}(\mathbf{x}, \mathbf{p})$ is differentiable with respect to $\mathbf{x}$ and $\mathbf{p}$ for all $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{p} \in \mathbb{R}^q$.

In section 5.1 we consider a model for which the operator $\mathbf{f}$ is a linear function of the model state. In this case we can re-write the model (2.1) in the form

$$\mathbf{x}_{k+1} = \mathbf{M}_k(\mathbf{p})\mathbf{x}_k \qquad k = 0, 1, \ldots \tag{2.2}$$

where the matrix $\mathbf{M}_k \in \mathbb{R}^{n \times n}$ is a non-singular matrix that depends non-linearly on the parameters $\mathbf{p}$.

For sequential assimilation, we start with a *background* state $\mathbf{x}_k^b \in \mathbb{R}^n$, with error $\boldsymbol{\varepsilon}_k^b \in \mathbb{R}^n$, that represents an *a priori* estimate of the true system state $\mathbf{x}_k$ at time t_k. This is a best guess estimate of the current system state obtained (for example) from a previous model forecast.

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

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \boldsymbol{\delta}_k, \qquad k = 0, 1, \ldots \tag{2.3}$$

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 and is commonly interpreted as a white noise sequence (Lewis et al. (2006)). The operator $\mathbf{h}_k : \mathbb{R}^n \longrightarrow \mathbb{R}^{r_k}$ is a nonlinear observation operator that maps from model to observation space. If we have direct measurements but at points that do not coincide with the model grid, $\mathbf{h}$ is simply an interpolation operator that interpolates the model variables 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.

2.2 The augmented system

The model (2.1) depends on parameters $\mathbf{p}$ whose values are imprecisely known. Sediment transport models, for example, are typically based on empirical formulae that use various parameterizations to characterise the physical properties of the sediment flux. We assume that these parameters remain constant over time, that is, they are not altered by the forecast model from one time step to the next. The evolution of the parameters can therefore be described by the equation

$$\mathbf{p}_{k+1} = \mathbf{p}_k, \qquad k = 0, 1, \ldots \ . \tag{2.4}$$

We define a new vector $\mathbf{w}$ by augmenting the standard model state $\mathbf{x}$ with the parameter vector $\mathbf{p}$

$$\mathbf{w} = \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} \in \mathbb{R}^{n+q}. \tag{2.5}$$

Combining (2.4) with the model for the evolution of the state (2.1) we can write the equivalent augmented system model as

$$\mathbf{w}_{k+1} = \tilde{\mathbf{f}}(\mathbf{w}_k), \tag{2.6}$$

where

$$\tilde{\mathbf{f}}(\mathbf{w}_k) = \begin{pmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) \\ \mathbf{p}_k \end{pmatrix}, \tag{2.7}$$

with $\tilde{\mathbf{f}} : \mathbb{R}^{n+q} \longrightarrow \mathbb{R}^{n+q}$.

We rewrite the equation for the observations (2.3) in terms of the augmented state vector as

$$\mathbf{y}_k = \tilde{\mathbf{h}}_k(\mathbf{w}_k) + \boldsymbol{\delta}_k, \tag{2.8}$$

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

$$\tilde{\mathbf{h}}(\mathbf{w}) = \tilde{\mathbf{h}} \begin{pmatrix} \mathbf{x} \\ \mathbf{p} \end{pmatrix} = \mathbf{h}(\mathbf{x}). \tag{2.9}$$

The aim of data assimilation is to combine the measured observations $\mathbf{y}_k$ with the augmented model predictions $\mathbf{w}_k^b$ to produce an updated augmented model state that most accurately describes the true augmented system state $\mathbf{w}_k^t$ at time t_k. This optimal estimate is called the *analysis* and is denoted $\mathbf{w}_k^a$. Note that the initial background state at t_0, $\mathbf{w}_0^b \in \mathbb{R}^{n+q}$, must include prior estimates of both the initial system state $\mathbf{x}_0$ and parameters $\mathbf{p}_0$. In addition to the updated state estimate, $\mathbf{x}_k^a$, the analysis $\mathbf{w}_k^a$ will also include updated estimates of the model parameters, $\mathbf{p}_k^a$ at each k.

3 Sequential data assimilation methods

A wide variety of data assimilation schemes exist (e.g. Kalnay (2003), Lewis et al. (2006)). In this study we combine ideas from the methods of 3D Var and the Kalman filter to produce a new hybrid scheme. Sections (3.1) and (3.2) give a brief overview of these two methods. Although we discuss their formulation specifically in terms of our augmented system we note that both schemes were orginally developed for basic state estimation. The equations are equivalent and can be derived by simply omitting the parameter vector from our descriptions.

3.1 3D Var

The 3D Var method (e.g. Courtier et al. (1998)) 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 $\mathbf{w}_k$ and the background state $\mathbf{w}_k^b$ and the observations $\mathbf{y}_k$,

$$J(\mathbf{w}_k) = (\mathbf{w}_k - \mathbf{w}_k^b)^{\mathbf{T}}\mathbf{B}_k^{-1}(\mathbf{w}_k - \mathbf{w}_k^b) + (\mathbf{y}_k - \tilde{\mathbf{h}}_k(\mathbf{w}_k))^T\mathbf{R}_k^{-1}(\mathbf{y}_k - \tilde{\mathbf{h}}_k(\mathbf{w}_k)). \tag{3.1}$$

The matrices $\mathbf{B}_k \in \mathbb{R}^{(n+q)\times(n+q)}$ and $\mathbf{R}_k \in \mathbb{R}^{r_k \times r_k}$ are the covariance matrices of the background and observation errors. These matrices represent the uncertainties of the background and observations and determine the relative weighting of $\mathbf{w}_k^b$ 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.

The analysis $\mathbf{w}_k^a$ satisfies the equation

$$\nabla J(\mathbf{w}_k^a) = 2\mathbf{B}_k^{-1}(\mathbf{w}_k^a - \mathbf{w}_k^b) - 2\tilde{\mathbf{H}}_k^T\mathbf{R}_k^{-1}(\mathbf{y}_k - \tilde{\mathbf{h}}_k(\mathbf{w}_k^a)) = 0, \tag{3.2}$$

where ∇J is the gradient of the cost function (3.1) with respect to $\mathbf{w}_k$, and the matrix $\tilde{\mathbf{H}}_k \in \mathbb{R}^{r_k\times(n+q)}$ represents the linearisation (or Jacobian) of the observation operator $\tilde{\mathbf{h}}_k$ evaluated at the background state $\mathbf{w}_k^b$.

When the observation operator, $\mathbf{h}_k$, is linear the minimum of (3.1) can be found exactly and the solution for the analysis can be written explicitly as

$$\mathbf{w}_k^a = \mathbf{w}_k^b + \mathbf{K}_k(\mathbf{y}_k - \tilde{\mathbf{H}}_k\mathbf{w}_k^b), \tag{3.3}$$

where

$$\tilde{\mathbf{h}}_k = \tilde{\mathbf{H}}_k \overset{\text{def}}{\equiv} \begin{pmatrix} \mathbf{H}_k & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{r_k\times(n+q)} \tag{3.4}$$

with $\mathbf{H}_k \in \mathbb{R}^{r_k\times n}$.

The operator $\mathbf{K} \in \mathbb{R}^{(n+q)\times r_k}$ is known as the gain matrix and is given by

$$\mathbf{K}_k = \mathbf{B}_k\tilde{\mathbf{H}}_k^T(\tilde{\mathbf{H}}_k\mathbf{B}_k\tilde{\mathbf{H}}_k^T + \mathbf{R}_k)^{-1}. \tag{3.5}$$

Equation (3.3) is known as the Optimal Interpolation formula or Best Linear Unbiased Estimator (BLUE) (Lewis et al. (2006)). When the dimension of the problem is small it is possible to calculate $\mathbf{K}$ explicitly

and compute the analysis directly from (3.3). The 3D Var method finds the analysis $\mathbf{w}_k^a$ numerically using a gradient descent algorithm to iterate to the minimising solution (Gill et al. (1981)). For systems of high dimension the OI method is impractical and it is more efficient to adopt a 3D Var approach.

The crucial difference between standard 3D Var and other schemes such as 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. 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.

3.2 The Kalman filter

The Kalman filter is also a sequential method. It was developed by Kalman (1960) and Kalman and Bucy (1961) and initially used in engineering applications. For a linear system, the Kalman filter 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 (Barnett and Cameron (1990), Jazwinski (1970)).

The main distinctions between the Kalman filter and 3D Var is 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 Kalman filter forecasts $\mathbf{B}$ forward, using knowledge of the quality of the current analysis to specify the covariances for the next assimilation step.

3.2.1 The Kalman filter predict and update equations

Below we present the Kalman filter algorithm for a discrete linear time-invariant model of the form

$$\mathbf{w}_{k+1} = \mathbf{F}\mathbf{w}_k \qquad k = 0, 1, \ldots$$

with observations linearly related to the state by the equations

$$\mathbf{y}_k = \tilde{\mathbf{H}}_k \mathbf{w}_k + \boldsymbol{\delta}_k. \tag{3.6}$$

Here $\mathbf{F} \in \mathbb{R}^{(n+q)\times(n+q)}$ is a constant, non-singular matrix describing the dynamic evolution of the state from time t_k to time t_{k+1} and $\tilde{\mathbf{H}}_k \in \mathbb{R}^{r_k \times (n+q)}$. We are assuming a perfect model (i.e. zero model error) but note that this is not a necessary assumption since the Kalman filter does allow for the inclusion of random model error (see for example, Martin et al. (1999)). The Kalman filter notation differs slightly from 3D Var: the background state vector $\mathbf{w}^b$ is replaced by the forecast vector $\mathbf{w}^f$ 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 consists of the following steps:

State forecast:

$$\mathbf{w}_{k+1}^f = \mathbf{F}\mathbf{w}_k^a \tag{3.7}$$

Error covariance forecast:

$$\mathbf{P}_{k+1}^f = \mathbf{F}\mathbf{P}_k^a\mathbf{F}^T \tag{3.8}$$

Kalman gain:

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1}^f \tilde{\mathbf{H}}_{k+1}^T (\tilde{\mathbf{H}}_{k+1}\mathbf{P}_{k+1}^f\tilde{\mathbf{H}}_{k+1}^T + \mathbf{R}_{k+1})^{-1} \tag{3.9}$$

Analysis:

$$\mathbf{w}_{k+1}^a = \mathbf{w}_{k+1}^f + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \tilde{\mathbf{H}}_{k+1}\mathbf{w}_{k+1}^f) \tag{3.10}$$

Analysis error covariance:

$$\mathbf{P}^a_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\tilde{\mathbf{H}}_{k+1})\mathbf{P}^f_{k+1}(\mathbf{I} - \mathbf{K}_{k+1}\tilde{\mathbf{H}}_{k+1})^T + \mathbf{K}_{k+1}\mathbf{R}_{k+1}\mathbf{K}^T_{k+1}. \tag{3.11}$$

If the Kalman gain $\mathbf{K}$ has been computed exactly this reduces to

$$\mathbf{P}^a_{k+1} = (\mathbf{I} - \mathbf{K}_{k+1}\tilde{\mathbf{H}}_{k+1})\mathbf{P}^f_{k+1}. \tag{3.12}$$

The optimality of the Kalman filter solution depends on the assumptions underlying these equations being accurate. Note that the analysis equation (3.10) and the definition of $\mathbf{K}$ (3.9) are the same as equations (3.3) and (3.5) for the BLUE with $\mathbf{B}_k = \mathbf{P}^f_k$.

3.3 The Extended Kalman Filter (EKF)

The Kalman filter theory can be generalised for the case where the system model and/ or observation operator are non-linear by linearising around a background state. This gives the extended Kalman filter (EKF) (Gelb (1974), Jazwinski (1970)). The steps of the EKF algorithm are the same as for the standard Kalman filter except that the state forecast (3.7) is made using the full non-linear model and the matrices $\mathbf{F}$ and $\tilde{\mathbf{H}}_k$ in equations (3.8) to (3.12) are replaced by the tangent linear model of the non-linear model forecast operator $\tilde{\mathbf{f}}$ and the tangent linear of the non-linear observation operator $\tilde{\mathbf{h}}_k$.

To summarise, we have

State forecast:

$$\mathbf{w}^f_{k+1} = \tilde{\mathbf{f}}(\mathbf{w}^a_k, \mathbf{p}^a_k) \tag{3.13}$$

Error covariance forecast:

$$\mathbf{P}^f_{k+1} = \mathbf{F}_k\mathbf{P}^a_k\mathbf{F}^T_k, \tag{3.14}$$

where

$$\mathbf{F}_k = \left.\frac{\partial\tilde{\mathbf{f}}}{\partial\mathbf{w}}\right|_{\mathbf{w}^a_k} = \left.\begin{pmatrix} \frac{\partial\mathbf{f}(\mathbf{z},\mathbf{p})}{\partial\mathbf{z}} & \frac{\partial\mathbf{f}(\mathbf{z},\mathbf{p})}{\partial\mathbf{p}} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}\right|_{\mathbf{z}^a_k,\mathbf{p}^a_k} \tag{3.15}$$

is the Jacobian of the augmented system forecast model evaluated at the current analysis state $\mathbf{w}^a_k$ (see appendix 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. The optimal analysis property of the standard linear Kalman filter no longer holds and the actual analysis error may differ considerably from that implied by equation (3.11).

The Kalman filter and EKF methods are compuationally much costlier than 3D Var; the updating of the error covariance matrices requires the equivalent of $O(n)$ model integrations, where n is dimension of the model state, plus adjoint and tangent linear models must be developed. If n is large the scheme becomes prohibitively expensive. Implementation of the full Kalman filter equations is therefore impracticable for systems of high dimension and in practice $\mathbf{P}^f$ 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) (Evensen (1994), Houtekamer and Mitchell (2005)) and the reduced rank Kalman filter (Fisher (1998)).

4 A hybrid approach

Although the technique of state augmentation is straightforward in theory, practical implementation of the approach 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. Since it is not possible to observe the parameters themselves, the parameter estimates will depend on the observations of the state variables. 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. For the augmented system, it is the cross covariances describing the correlations between errors in the parameters and errors in the model state estimate that pass information from the observed variables to update the estimates of the unobserved parameters. This is a crucial point; if these cross covariances are inappropriately modelled the quality of the parameter estimates will be affected. Since the correct error statistics of the system are generally unknown we have to approximate them in some manner. Constructing a realistic representation of the background error covariances is one of the key challenges of data assimilation.

Previous work (Smith et al. (2008), Smith et al. (2009a)) indicated that whilst the assumption of static covariances made by the 3D Var algorithm is sufficient for state estimation it is insufficient for parameter estimation as it does not provide an adequate representation of the state-parameter cross covariances. In order to reliably estimate the parameters the cross-covariances between the parameters and the state need to evolve with the model.

We have already noted that using methods such as the Kalman filter to explicitly propagate the covariances is computationally expensive and requires the construction of adjoint and tangent linear models. Motivated by a desire for a low cost, uncomplicated alternative we have combined ideas from 3D-Var and the EKF to produce a new hybrid assimilation scheme that captures the flow dependent nature of the state-parameter cross covariances without 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. We give details of the formulation of this new approach below.

4.1 Formulation

We can partition the forecast error covariance matrix (3.14) as follows

$$\mathbf{P}_k^f = \begin{pmatrix} \mathbf{P}^f_{\mathbf{x}\mathbf{x}_k} & \mathbf{P}^f_{\mathbf{x}\mathbf{p}_k} \\ (\mathbf{P}^f_{\mathbf{x}\mathbf{p}_k})^T & \mathbf{P}^f_{\mathbf{p}\mathbf{p}_k} \end{pmatrix}. \tag{4.1}$$

Here $\mathbf{P}^f_{\mathbf{x}\mathbf{x}_k} \in \mathbb{R}^{n\times n}$ is the forecast error covariance matrix for the state vector $\mathbf{x}_k$ at time t_k, $\mathbf{P}^f_{\mathbf{p}\mathbf{p}_k} \in \mathbb{R}^{q\times q}$ is the forecast error covariance matrix for the parameter vector $\mathbf{p}_k$ and $\mathbf{P}^f_{\mathbf{x}\mathbf{p}_k} \in \mathbb{R}^{n\times q}$ is the covariance matrix for the cross correlations between the forecast errors in the state and parameter vectors.

We use the EKF equations (3.13)-(3.15) as a guide and consider the form of the forecast error covariance for a single step of the filter. Suppose we start at time t_k with analysis error covariance matrix

$$\mathbf{P}_k^a = \begin{pmatrix} \mathbf{P}^a_{\mathbf{x}\mathbf{x}_k} & \mathbf{P}^a_{\mathbf{x}\mathbf{p}_k} \\ (\mathbf{P}^a_{\mathbf{x}\mathbf{p}_k})^T & \mathbf{P}^a_{\mathbf{p}\mathbf{p}_k} \end{pmatrix} \in \mathbb{R}^{(n+q)\times(n+q)}. \tag{4.2}$$

If we denote

$$\mathbf{M}_k = \left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{p})}{\partial \mathbf{z}}\right|_{\mathbf{x}_k^a,\mathbf{p}_k^a} \qquad \text{and} \qquad \mathbf{N}_k = \left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{p})}{\partial \mathbf{p}}\right|_{\mathbf{x}_k^a,\mathbf{p}_k^a}, \tag{4.3}$$

where $\mathbf{M}_k \in \mathbb{R}^{m\times m}$ and $\mathbf{N}_k \in \mathbb{R}^{m\times q}$, we can write the linearised augmented forecast model (3.15) as

$$\mathbf{F}_k = \begin{pmatrix} \mathbf{M}_k & \mathbf{N}_k \\ \mathbf{0} & \mathbf{I} \end{pmatrix}. \tag{4.4}$$

The error covariance forecast (3.14) is then given by

$$\begin{aligned}
\mathbf{P}^f_{k+1} &= \begin{pmatrix} \mathbf{M}_k & \mathbf{N}_k \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{P}^a_{\mathbf{xx}_k} & \mathbf{P}^a_{\mathbf{xp}_k} \\ (\mathbf{P}^a_{\mathbf{xp}_k})^T & \mathbf{P}^a_{\mathbf{pp}_k} \end{pmatrix} \begin{pmatrix} \mathbf{M}_k^T & \mathbf{0} \\ \mathbf{N}_k^T & \mathbf{I} \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{M}_k\mathbf{P}^a_{\mathbf{xx}_k} + \mathbf{N}_k(\mathbf{P}^a_{\mathbf{xp}_k})^T & \mathbf{M}_k\mathbf{P}^a_{\mathbf{xp}_k} + \mathbf{N}_k\mathbf{P}^a_{\mathbf{pp}_k} \\ (\mathbf{P}^a_{\mathbf{xp}_k})^T & \mathbf{P}^a_{\mathbf{pp}_k} \end{pmatrix} \begin{pmatrix} \mathbf{M}_k^T & \mathbf{0} \\ \mathbf{N}_k^T & \mathbf{I} \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{M}_k\mathbf{P}^a_{\mathbf{xx}_k}\mathbf{M}_k^T + \mathbf{N}_k(\mathbf{P}^a_{\mathbf{xp}_k})^T\mathbf{M}_k^T + \mathbf{M}_k\mathbf{P}^a_{\mathbf{xp}_k}\mathbf{N}_k^T + \mathbf{N}_k\mathbf{P}^a_{\mathbf{pp}_k}\mathbf{N}_k^T & \mathbf{M}_k\mathbf{P}^a_{\mathbf{xp}_k} + \mathbf{N}_k\mathbf{P}^a_{\mathbf{pp}_k} \\ (\mathbf{P}^a_{\mathbf{xp}_k})^T\mathbf{M}_k^T + \mathbf{P}^a_{\mathbf{pp}_k}\mathbf{N}_k^T & \mathbf{P}^a_{\mathbf{pp}_k} \end{pmatrix}
\end{aligned} \tag{4.5}$$

We do not want to compute the whole matrix (4.5) at every time step so we make some simplifying assumptions:

We substitute the state forecast error covariance matrix $\mathbf{P}^f_{\mathbf{xx}_k}$ with a standard 3D Var fixed approximation

$$\mathbf{P}^f_{\mathbf{xx}_k} = \mathbf{B}_{\mathbf{xx}} \qquad \text{for all } k. \tag{4.6}$$

We assume that the parameter error covariance matrix is also fixed

$$\mathbf{P}^f_{\mathbf{pp}_k} = \mathbf{B}_{\mathbf{pp}} \qquad \text{for all } k, \tag{4.7}$$

Finally, we assume that the state-parameter cross covariances are initially zero. This leads us to propose the following approximation for the augmented forecast error covariance matrix

$$\mathbf{B}_{k+1} = \begin{pmatrix} \mathbf{B}_{\mathbf{xx}} & \mathbf{N}_k\mathbf{B}_{\mathbf{pp}} \\ \mathbf{B}_{\mathbf{pp}}\mathbf{N}_k^T & \mathbf{B}_{\mathbf{pp}} \end{pmatrix}. \tag{4.8}$$

In other words, all elements of the background error covariance matrix (4.8) are kept fixed except the state-parameter cross covariances

$$\mathbf{B}_{\mathbf{xp}_{k+1}} = \mathbf{N}_k\mathbf{B}_{\mathbf{pp}}, \tag{4.9}$$

which are updated at each new analysis time by recalculating the matrix $\mathbf{N}_k$, where $\mathbf{N}_k$ is the Jacobian of the forecast model with respect to the parameters, as defined in equation (4.3).

The analysis $\mathbf{w}^a_k$ is found by substituting the matrix (4.8) into the 3D Var cost function (3.1) and minimising. Although in practice the minimum is found numerically using a gradient descent method, the analytical form for the solution (3.3) is useful for understanding the role of the state-parameter cross-covariances.

For the experiments presented in this report we use direct observations taken at fixed locations. We can therefore write

$$\tilde{\mathbf{H}}_k = \tilde{\mathbf{H}} \equiv (\mathbf{H}\ \mathbf{0}) \qquad \text{for all } k. \tag{4.10}$$

The observation operator $\mathbf{H} \in \mathbb{R}^{r\times n}$ is a constant matrix and the number of observations $r_k = r$ is the same for all k. We note that this is not a necessary assumption and is made purely for ease of illustration and computation. A non-linear observation operator would have no effect on the formulation of our method and should be relatively straight forward to implement in practice.

The gain matrix (3.5) can now be written as

$$\begin{aligned}
\mathbf{K}_k &= \mathbf{B}_k\tilde{\mathbf{H}}^T(\tilde{\mathbf{H}}\mathbf{B}_k\tilde{\mathbf{H}}^T + \mathbf{R}_k)^{-1} \\
&\stackrel{\text{def}}{=} \begin{pmatrix} \mathbf{K}_{\mathbf{x}_k} \\ \mathbf{K}_{\mathbf{p}_k} \end{pmatrix}.
\end{aligned} \tag{4.11}$$

Separating (3.3) into state and parameter parts gives

$$\mathbf{x}^a_k = \mathbf{x}^b_k + \mathbf{K}_{\mathbf{x}_k}(\mathbf{y}_k - \mathbf{H}\mathbf{x}^b_k) \tag{4.12}$$

$$\mathbf{p}^a_k = \mathbf{p}^b_k + \mathbf{K}_{\mathbf{p}_k}(\mathbf{y}_k - \mathbf{H}\mathbf{x}^b_k), \tag{4.13}$$

where the gain matrices (4.11) are given by

$$\mathbf{K}_{\mathbf{x}_k} = \mathbf{B}_{\mathbf{xx}}\mathbf{H}^T(\mathbf{H}\mathbf{B}_{\mathbf{xx}}\mathbf{H}^T + \mathbf{R}_k)^{-1} \tag{4.14}$$

$$\mathbf{K}_{\mathbf{p}_k} = \mathbf{B}_{\mathbf{pp}}\mathbf{N}_{k-1}^T\mathbf{H}^T(\mathbf{H}\mathbf{B}_{\mathbf{xx}}\mathbf{H}^T + \mathbf{R}_k)^{-1}. \tag{4.15}$$

Note that since we are assuming that $\mathbf{B_{xx}}$ and $\mathbf{H}$ are constant matrices, the state gain $\mathbf{K}_{\mathbf{x}_k} = \mathbf{K_x}$ is fixed for all time. The analysis equation for the state vector (4.12) is therefore the same as would be derived if the 3D Var method was being used for state estimation only.

The innovation vector $(\mathbf{y}_k - \mathbf{H}\mathbf{x}_k^b)$ is exactly the same in equations (4.12) and (4.13), as is the expression inside the inverse for the state and parameter gain matrices (4.14) and (4.15). 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. This is determined by our choice of $\mathbf{B_{xx}}$ and $\mathbf{B_{xp}}$. We have already stated that, in order to reliably estimate the model parameters, the matrix $\mathbf{B_{xp}}$ must adequately describe the relationship between the errors in the state estimate and the errors in the parameters. Our proposed approximation to $\mathbf{B_{xp}}$ (4.9) does this by combining the relationship between the errors in the parameters (described by $\mathbf{B_{pp}}$) with the way changes in the parameters affect the forecast model (described by $\mathbf{N}_k$).

5 The models

We demonstrate this new hybrid approach using three simple models: a single parameter 1D linear advection model, a two parameter linear damped oscillating system, and the three parameter non-linear Lorenz 63 equations (Lorenz (1963), Sparrow (1982)).

The scheme has been tested by carrying out a series of identical twin experiments using pseudo-observations with a range of temporal and (where applicable) spatial frequencies. The aim was to assess the applicability of our proposed approach to modelling the background error covariances and determine whether it is sufficient to enable the recovery of uncertain model parameters in range of dynamical system models. In each case we specify a 'true' solution which is generated by running the model from a given initial condition with set values for the model parameters. This solution is used to provide observations for the assimilation and also to evaluate the performance of the scheme in terms of estimating the state variables. The model is then re-run with the data assimilation, starting from a perturbed initial state and with incorrect estimates of the parameters.

In this section we give details of the models and derive estimates for the state-parameter cross covariance matrix for each specific case. A brief description of the experimental design is also given, followed by the results.

5.1 Linear advection

We first consider the one-dimensional linear advection equation,

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \tag{5.1}$$

where a is the advection velocity or wave speed.

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

$$u(x,0) = u_0(x), \qquad -\infty < x < \infty, \tag{5.2}$$

the solution of (5.1) at time $t \geq 0$ is simply (LeVeque (1992))

$$u(x,t) = u(x_0,0) = u_0(x - at), \tag{5.3}$$

where $x_0 = x(0)$ and $x(t) = x_0 + at$.

The solution (5.3) has the property that it preserves its initial shape u_0. As time evolves, the initial data propagates undistorted at constant speed a to the right (if $a > 0$) or left (if $a < 0$).

We choose to solve (5.1) for $a > 0$ on the finite spatial domain $x \in [\,0, 2\,]$, with periodic boundary conditions

$$u(0,t) = u(2,t), \tag{5.4}$$

and initial data given by the Gaussian function

$$u(x,0) = \begin{cases} 0 & x < 0.01 \\ e^{-\frac{(x-0.25)^2}{0.01}} & 0.01 < x < 0.5 \\ 0 & x \geq 0.5 \end{cases} \tag{5.5}$$

For our assimilation scheme we require $u(x,t)$ at discrete points (x_j, t_k). We assume that $u(x,t)$ is continuous and differentiable and discretise using the upwind scheme (LeVeque (1992)),

$$u_j^{k+1} = u_j^k - a\frac{\Delta t}{\Delta x}(u_j^k - u_{j-1}^k), \qquad j = 1, 2, \ldots, n \qquad k = 0, 1, \ldots, T \tag{5.6}$$

with boundary conditions

$$u_0^k = u_n^k \tag{5.7}$$

where $u_j^k \approx u(x_j, t_k)$ and $x_j = j\Delta x$, $t_k = k\Delta t$, where Δx is the spatial grid spacing and Δt is the model time step.

Denoting $\mu = \frac{\Delta t}{\Delta x}$, we can rewrite (5.6) as

$$u_j^{k+1} = (1 - a\mu)u_j^k + a\mu u_{j-1}^k. \tag{5.8}$$

The upwind scheme is first order accurate and stable provided that the CFL condition $a\mu \leq 1$ is satisfied. To ensure that the model remains stable during the assimilation we set $\Delta x = 0.01$, $\Delta t = 0.01$ and assume that a is known to be on the interval $0 \leq a \leq 1$.

The forecast model (5.8) can be expressed as the matrix system

$$\mathbf{u}_{k+1} = \mathbf{A}\mathbf{u}_k \tag{5.9}$$

where $\mathbf{u}_k \in \mathbb{R}^n$ is the model state at time t_k and $\mathbf{A}$ is a (constant) $n \times n$ matrix, given by

$$\mathbf{A} = \mathbf{A}(a) \begin{pmatrix} (1-a\mu) & 0 & & & a\mu \\ a\mu & (1-a\mu) & 0 & \ldots & \\ & \ddots & \ddots & \ddots & \\ & & & & 0 \\ 0 & \ldots & 0 & a\mu & (1-a\mu) \end{pmatrix}, \tag{5.10}$$

Since the advection velocity a is constant, we have

$$a_{k+1} = a_k, \tag{5.11}$$

cf. equation 2.4.

Setting

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{u}_k \\ a_k \end{pmatrix}, \tag{5.12}$$

we can combine (5.9) and (5.11) to give the augmented system model

$$\mathbf{w}_{k+1} = \mathbf{f}(\mathbf{w}_k) \tag{5.13}$$

$$= \begin{pmatrix} \mathbf{A}(a_k) & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{u}_k \\ a_k \end{pmatrix} \tag{5.14}$$

Note that the matrix $\mathbf{A}$ has been replaced by the parameter dependent matrix $\mathbf{A}(a_k)$. Although the true $\mathbf{A}$ is constant, the forecast model at time t_k will depend on the current estimate a_k of the true advection velocity a; this will vary as a_k is updated by the assimilation process.

5.1.1 State-parameter cross covariance

For the cross covariances between the errors in the model state $\mathbf{u}$ and parameter a we need to calculate the matrix $\mathbf{N}_k$; the Jacobian of the forecast model with respect to the model parameters. For the linear advection model this is given by

$$\mathbf{N}_k = \left. \frac{\partial \left(\mathbf{A}_k \mathbf{u}_k \right)}{\partial a_k} \right|_{\mathbf{u}_k^a, a_k^a}. \tag{5.15}$$

From (5.10) we have

$$\frac{\partial \mathbf{A}_k}{\partial a_k} = \begin{pmatrix} -\mu & 0 & & & \mu \\ \mu & -\mu & 0 & \dots & \\ & \ddots & \ddots & \ddots & \\ & & & & 0 \\ 0 & \dots & 0 & \mu & -\mu \end{pmatrix}. \tag{5.16}$$

which is constant for all a_k.

The matrix $\mathbf{N}_k$ is a $n \times 1$ vector with elements N_j

$$N_j = \frac{\partial u_j^{k+1}}{\partial a_k} = -\mu (u_j^k - u_{j-1}^k), \qquad j = 1, \dots, n. \tag{5.17}$$

Since we only have a single unknown parameter, the parameter vector $\mathbf{p}^b$ is scalar and

$$\mathbf{B_{pp}} = \sigma_a^2, \tag{5.18}$$

where σ_a^2 is the parameter error variance.

Combining (5.17) and (5.18) we have

$$b_{up}^{k+1}(j) = -\sigma_a^2 \, \mu (u_j^k - u_{j-1}^k), \qquad j = 1, \dots, n. \tag{5.19}$$

where $b_{up}^{k+1}(j)$ represents the cross covariance between the parameter error ε_a and element j of the state background error vector $\boldsymbol{\varepsilon}_u$ at time t_{k+1}.

5.1.2 Assimilation experiments

We set the true advection velocity to be $a = 0.5$. The initial parameter estimate a_0 is generated by adding Gaussian random noise with zero mean and variance $\sigma_a^2 = 0.1$ to this value. This corresponds to an error variance of 20%. The assimilation process is carried out sequentially. Observations are generated by sampling the analytic solution (5.3) on a regularly spaced grid and are assimilated at regular time intervals. The state background error covariance matrix $\mathbf{B_{xx}}$ is kept fixed, as is the parameter background error covariance matrix $\mathbf{B_{pp}}$. The state-parameter cross covariance matrix $\mathbf{B_{xp}}$ is recalculated at each new assimilation time as described above. At the end of each assimilation cycle the model parameters 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 analysis time.

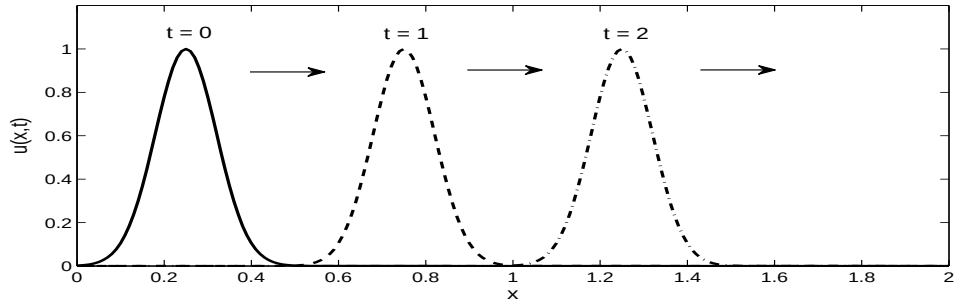

Figure 5.1: Linear advection model: true solution $u(x,t)$

We assume that the general shape of the initial data is known. The initial background state $\mathbf{u}_0^b$ is therefore taken to be of the form (5.5) but is rescaled so that it is slightly shorter, narrower and in a different starting position to the true initial state. The state background error covariance matrix $\mathbf{B_{xx}}$ is kept fixed, as is the parameter background error covariance matrix $\mathbf{B_{pp}}$. To characterise the state background errors we use the isotropic correlation function (Rodgers (2000))

$$b_{ij} = \sigma_b^2 \rho^{|i-j|}, \qquad i,j = 1,\ldots,n, \tag{5.20}$$

where element b_{ij} defines the covariance between components i and j of the state background error vector $\boldsymbol{\varepsilon}_u = \mathbf{u}^b - \mathbf{u}^t$. Here $\rho = \exp(-\Delta x/L)$ where Δx is the model grid spacing and L is a correlation length scale that is adjusted empirically[1], and σ_b^2 is the state background error variance which we set equal to 0.05.

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

$$\mathbf{R}_k = \mathbf{R} = \sigma_o^2 \mathbf{I}, \qquad \mathbf{I} \in \mathbb{R}^{r\times r}, \tag{5.21}$$

where r is the number of observations and σ_o^2 is the observation error variance.

We run the model on the domain $x \in [0,2]$ so that the dimension of our system is $n+q = 201$. The true solution is shown in figure 5.1. We adopt a 3D Var approach and minimise the cost function (3.1) iteratively using a quasi-Newton descent algorithm (Gill et al. (1981)).

5.1.3 Results

Perfect observations

The experiments were carried out using a range of both over and under estimated initial a values, with varying spatial and temporal observation combinations. The convergence and accuracy of the parameter estimates and depends on a number of factors such as the quality of the initial background guess, the location and spatial frequency of the observations and the time between successive assimilations.

Figure 5.2(a) shows the effect of varying the spatial frequency of observations for the initial estimate $a_0 = 0.87116$. Observations are assimilated every $10\Delta t$, with $\sigma_o^2 = 0.01$ and grid spacings between observations range from every $2\Delta x$ to every $25\Delta x$. The scheme performs extremely well. For observations taken every $2\Delta x$, $10\Delta x$ and $25\Delta x$ the scheme recovers the true value of a to a high level of accuracy with only slight differences in the rate of convergence. The quality of the state analysis is also high. There is a noticable difference when the spacing of observations is decreased to every $50\Delta x$ and then further again to every $100\Delta x$. Here the estimates take much longer to stabilise. With observations every $100\Delta x$ the a estimate gets close to but never quite settles on the true value of a, even when the assimilation period is extended beyond that shown. The state analysis is also less accurate.

Figure 5.2(b) shows the effect of varying the temporal frequency of the observations for the same starting estimate $a_0 = 0.87116$. Observations are taken every $5\Delta x$, and assimilated at intervals of $5\Delta t$, $10\Delta t$, $25\Delta t$ and $50\Delta t$. The results are similar to the previous experiment; there are only small differences

[1] for these experiments it is set at twice the current observation spacing.

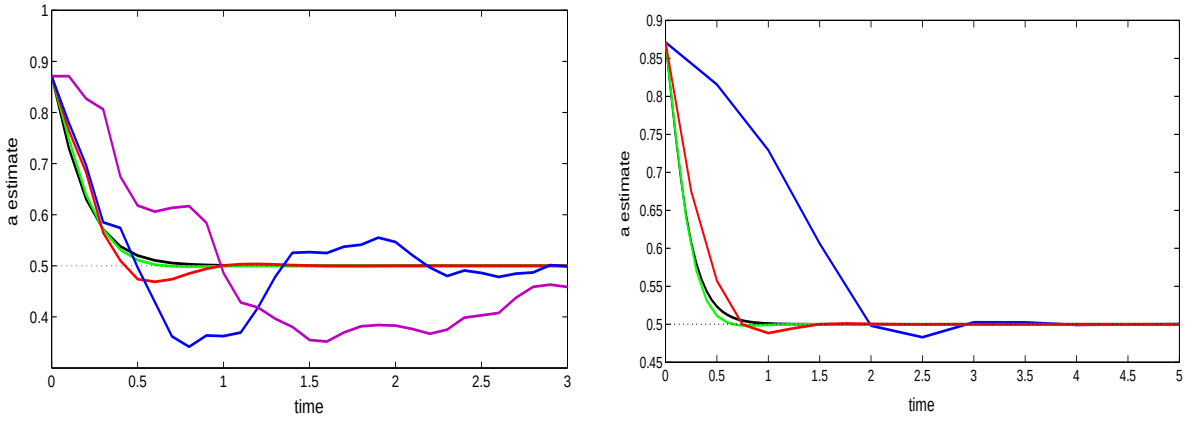

Figure 5.2: **Perfect observations:** parameter updates for initial estimate $a = 0.87116$. **(a)** Varying the spatial frequency of observations: solid black line - observations at $2\Delta x$ intervals; solid green line - observations at $5\Delta x$ intervals; solid red line - observations at $10\Delta x$ intervals; solid blue line - observations at $25\Delta x$ intervals; solid purple line - observations at $50\Delta x$ intervals. **(b)** Varying the temporal frequency of observations: 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 $50\Delta t$.

visible when the time between successive assimilations is increased from every $5\Delta t$ to $10\Delta t$ to $25\Delta t$. For observations every $50\Delta t$ the a estimate takes slightly longer to converge and if this period is doubled again to $100\Delta t$ the scheme completely fails to recover a.

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 are taken at $5\Delta x$ intervals and assimilated every $10\Delta t$. Figure 5.3(a) shows the parameter a estimates produced for error variance increasing from $\sigma_o^2 = 0.0001$ 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 analysis and parameter estimates are also noisy. When $\sigma_o^2 = 0.1$ the state analysis is particularly messy especially around the tails of the curve where u values are close to zero. The amplitude of oscillations in a increase as σ_o^2 is increased. The oscillations are, however, approximately centered around the true a 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 figure 5.3(b). Here, the above experiment has been repeated but with the a estimates being averaged over a moving time window of 20 timesteps. Note that to allow time for the scheme to settle we omit the early estimates and begin the averging at $t = 1.5$.

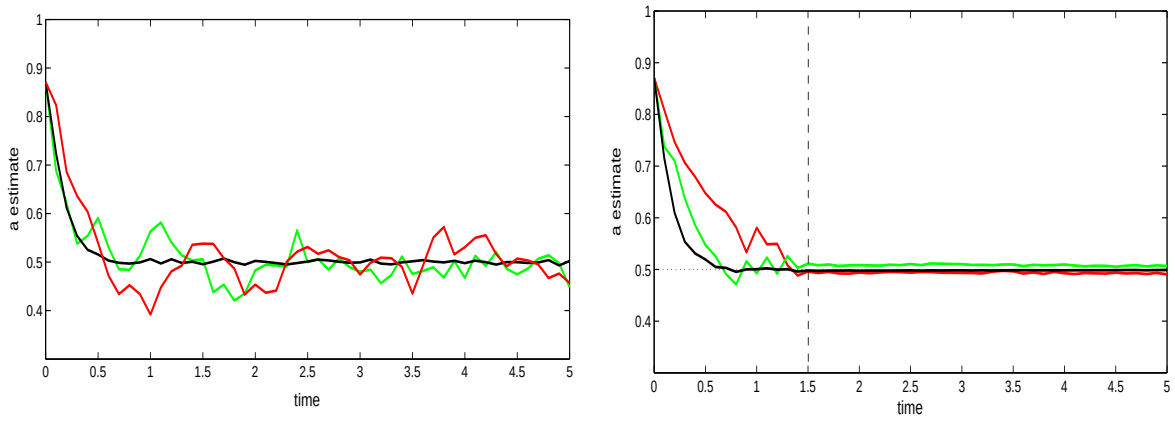

Figure 5.3: **Imperfect observations:** parameter updates for initial estimate $a = 0.87116$ **(a)** unaveraged estimates, and **(b)** time averaged estimates: solid black line $\sigma_o^2 = 0.001$; solid green line $\sigma_o^2 = 0.01$; solid red line $\sigma_o^2 = 0.1$.

5.2 Nonlinear oscillator

Our second test model is an unforced, damped non-linear oscillator given by the second order ordinary differential equation

$$\ddot{x} + d\dot{x} + mx + x^3 = 0. \tag{5.22}$$

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

Equation (5.22) is often referred to as the Duffing equation or Duffing oscillator (Duffing (1918)). It arises in a variety of applications and in a number of different forms. For a more detailed discussion see e.g. Guckenheimer and Holmes (1986), Wiggins (1990), Thompson and Stewart (1986). For $d, m > 0$, the form (5.22) 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 (Stoker (1966)).

Equation (5.22) can be written as the first order system

$$\begin{aligned} \dot{x} &= y, \\ \dot{y} &= -(mx + x^3 + dy). \end{aligned} \tag{5.23}$$

The nature of the solution of this system varies greatly depending on the values of the parameters. For $d, m > 0$ the sytem (5.23) has a single stable equilibrium at $(x, \dot{x}) = (0, 0)$.

We solve (5.23) numerically using a second order Runge-Kutta method. Discretising the system (5.23) gives the following set of equations

$$x_{k+1} = \left(\Delta t - d\frac{\Delta t^2}{2}\right) y_k + \left(1 + m\frac{\Delta t^2}{2} - \frac{\Delta t^2}{2}x_k^2\right) x_k \tag{5.24}$$

$$\begin{aligned} y_{k+1} = {} & \left(1 - d\Delta t + m\frac{\Delta t^2}{2} + d^2\frac{\Delta t^2}{2}\right) y_k + \left(m\Delta t - dm\frac{\Delta t^2}{2} + (d\frac{\Delta t^2}{2} - \frac{\Delta t}{2})x_n^2\right) x_k \ldots \\ & -\frac{\Delta t}{2}\left(x_k + \Delta t y_k\right)^3 \end{aligned} \tag{5.25}$$

where the model time step $\Delta t = 0.1$.

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

$$\mathbf{p}_k = \begin{pmatrix} d_k \\ m_k \end{pmatrix}. \tag{5.26}$$

Since we are assuming that they are constant, we can write

$$\mathbf{p}_{k+1} = \mathbf{p}_k, \tag{5.27}$$

(cf. equation 2.4).

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

$$\mathbf{x}_k = \begin{pmatrix} x_k \\ y_k \end{pmatrix}, \tag{5.28}$$

gives the augmented state vector

$$\mathbf{w}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{p}_k \end{pmatrix}$$

(cf. equation 2.5).

This allows us to write (5.24)-(5.25) and (5.27) as the equivalent augmented system

$$\begin{aligned} \mathbf{w}_{k+1} &= \tilde{\mathbf{f}}(\mathbf{w}_k) \\ &= \begin{pmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{x}_k \\ \mathbf{p}_k \end{pmatrix}. \end{aligned} \tag{5.29}$$

5.2.1 State-parameter cross covariance

For the oscillating system the Jacobian of the forecast model with respect to the parameters is defined as

$$\begin{aligned} \mathbf{N}_k &= \left(\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial d_k} \quad \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial m_k} \right) \Bigg|_{\mathbf{x}_k^a, \mathbf{p}_k^a} \\ &= \begin{pmatrix} \frac{\partial x_{k+1}}{\partial d_k} & \frac{\partial x_{k+1}}{\partial m_k} \\ \frac{\partial y_{k+1}}{\partial d_k} & \frac{\partial y_{k+1}}{\partial m_k} \end{pmatrix} \Bigg|_{\mathbf{x}_{k+1}^b, \mathbf{p}_k^a} \end{aligned} \tag{5.30}$$

From (5.24) and (5.25) we have

$$\frac{\partial x_{k+1}}{\partial d_k} = \frac{-\Delta t^2}{2} y_k, \tag{5.31}$$

$$\frac{\partial x_{k+1}}{\partial m_k} = \frac{\Delta t^2}{2} x_k \tag{5.32}$$

and

$$\frac{\partial y_{k+1}}{\partial d_k} = \left(d\Delta t^2 - \Delta t\right) y_k + \left(x_k^2 - m\right) \frac{\Delta t^2}{2} x_k \tag{5.33}$$

$$\frac{\partial y_{k+1}}{\partial m_k} = \frac{\Delta t^2}{2} y_k + \left(\Delta t - d\frac{\Delta t^2}{2}\right) x_k. \tag{5.34}$$

We assume that d and m are uncorrelated and set

$$\mathbf{B_{pp}} = \begin{pmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_m^2 \end{pmatrix}, \tag{5.35}$$

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

The state-parameter cross covariance matrix is then given by

$$\begin{aligned} \mathbf{B_{xp}}_{k+1} &= \mathbf{N}_k \mathbf{B_{pp}} \\ &= \left(\sigma_d^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial d_k} \quad \sigma_m^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial m_k} \right) \Bigg|_{\mathbf{x}_k^a, \mathbf{p}_k^a}. \end{aligned} \tag{5.36}$$

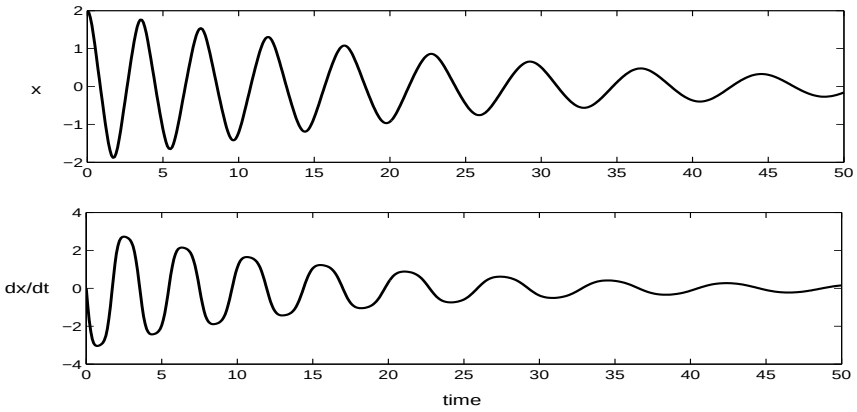

Figure 5.4: Damped, unforced nonlinear oscillator: computed numerical solution for x and y.

5.2.2 Assimilation experiments

We define the 'true' solution to be that given by the discretised equations (5.23) - (5.24) with initial displacement $x_0 = 2.0$ and initial velocity $y_0 = 0.0$ and parameter values $d = 0.1$ and $m = 0.5$. Observations of both x and y are taken from this solution and assimilated at regular time intervals. The true solution for x and y is shown in figure 5.4. 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 at

$$\mathbf{B_{xx}} = \sigma_b^2 \mathbf{I}, \qquad \mathbf{I} \in \mathbb{R}^{2\times 2}. \tag{5.37}$$

For the observation error covariance matrix we use

$$\mathbf{R} = \sigma_o^2 \mathbf{I}, \qquad \mathbf{I} \in \mathbb{R}^{2\times 2}. \tag{5.38}$$

We generate an initial estimates d_0 and m_0 for the parameters d and m by adding random noise with error variance $\sigma_d^2 = 0.01$ and $\sigma_m^2 = 0.05$ respectively to the 'true' values. Since in this case the dimension of augmented state is small ($n + q = 4$) we compute the analysis directly from equation (3.3).

5.2.3 Results

Perfect observations

Figures 5.5(a) and (b) show the parameter d and m updates for initial estimates $d_0 = 0.056641$ and $m_0 = 0.74804$ with observations of x and y taken at varying temporal frequencies. The observation error variance is set at $\sigma_o^2 = 0.01$. The scheme manages to retrieve both d and m to a good level of accuracy for observation intervals up to every $25\Delta t$. In this example, the estimates produced using an observation interval of $25\Delta t$ are actually slightly better than when observations are taken every $10\Delta t$. We see a big increase in error when the observation frequency is decreased to $50\Delta t$; here the d and m estimates are almost twice their true values. A particular difficultly with this system is that the solution decays to zero quite quickly and so there is only a limited amount of time in which the assimilation scheme is of use. When observations are very infrequent there are not enough observations taken before the system becomes very damped and this limits the ability of our scheme to identify the true parameter values.

Noisy observations

The above experiments were repeated using imperfect observations. Figures 5.6(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 5.1.3 above. As with the linear advection model, the state and parameter estimates are both noisy. The impact of noise on the observations is greatest when observations are assimilated every Δt.

Here, the noisy observations are too frequent and the model does not have sufficient time to adjust to the new parameter value before the next input of data. For intervals of $5\Delta t$ to $25\Delta t$ the parameter estimates actually improve as the frequency of the observations is decreased.

Figure 5.7(a) and (b) show the effect of averaging the parameter estimates over a moving time window of 50 timesteps with averaging starting at $t = 20$. With the exception of the case where observations are taken at every timestep, the accuracy of the smoothed parameter estimates is similar to the perfect observations case and possibly even better for the parameter m. If the observation error variance is increased to $\sigma_o^2 = 0.1$ there is a noticeable difference in the quality of the state analysis but we are still able to obtain reasonable parameter estimates.

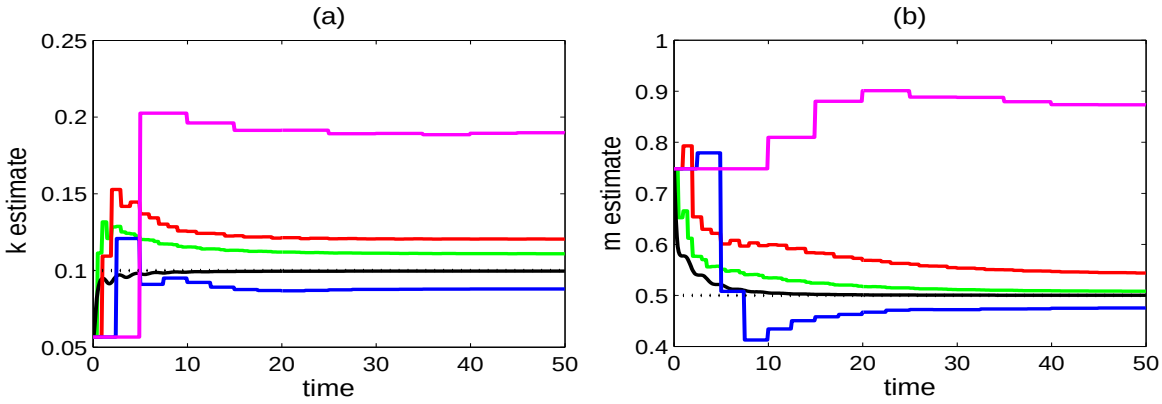

Figure 5.5: **Perfect observations (a)** Parameter d updates for initial estimate $d_0 = 0.056641$, **(b)** Parameter m updates for initial estimate $m_0 = 0.74804$: solid black line - observations every Δt; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid purple line - observations every $50\Delta t$.

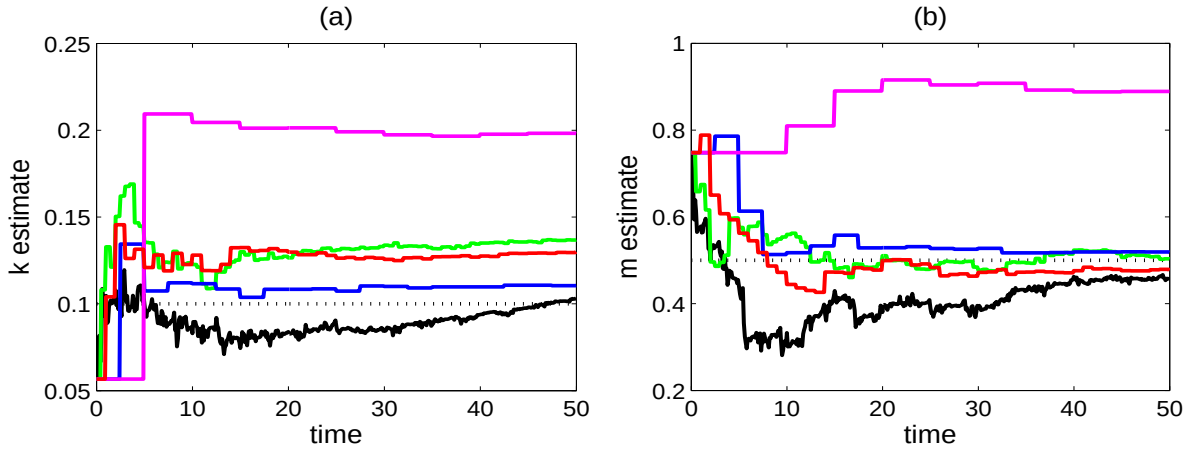

Figure 5.6: **Imperfect observations** $\sigma_o^2 = 0.01$ **(a)** Parameter d updates for initial estimate $d_0 = 0.056641$, **(b)** Parameter m updates for initial estimate $m_0 = 0.74804$: solid black line - observations every Δt; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid purple line - observations every $50\Delta t$.

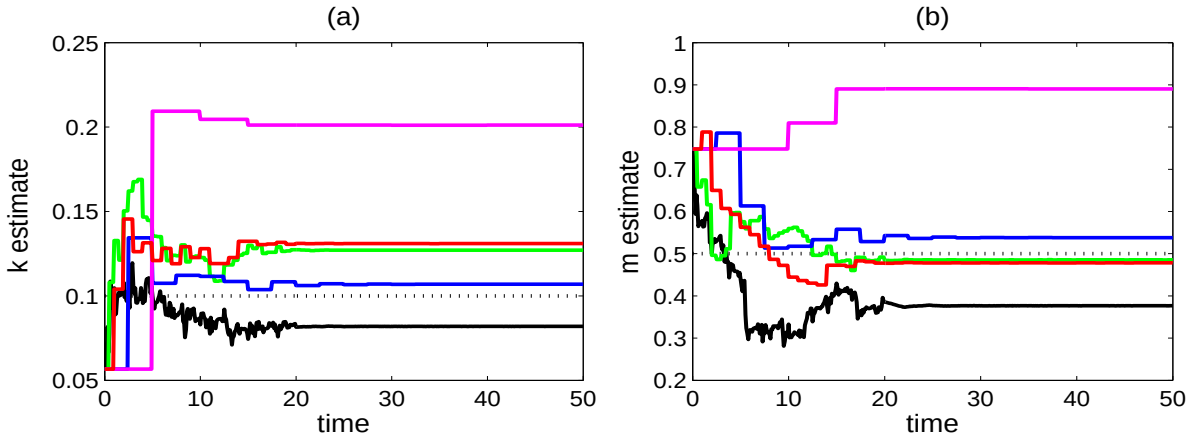

Figure 5.7: **Imperfect observations** $\sigma_o^2 = 0.1$ **(a)** Averaged parameter d updates for initial estimate $d_0 = 0.056641$, **(b)** Parameter m updates for initial estimate $m_0 = 0.74804$: solid black line - observations every Δt; solid green line - observations every $5\Delta t$; solid red line - observations every $10\Delta t$; solid blue line - observations every $25\Delta t$; solid purple line - observations every $50\Delta t$.

5.3 Lorenz 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 (Lorenz (1963))

$$\dot{x} = -\sigma(x - y)\,, \tag{5.39}$$
$$\dot{y} = \rho x - y - xz\,, \tag{5.40}$$
$$\dot{z} = xy - \beta z\,, \tag{5.41}$$

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 examinining the properties of data assimilation methods when applied to highly nonlinear dynamical systems.

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 giving rise to chaotic solutions (Sparrow (1982)).

For this system we adapt a pre-existing Matlab routine written by M.J Martin[2] (Martin et al. (1999), Martin (2000)). The program solves equations (5.39)-(5.41) numerically using a second order Runge-Kutta method (see e.g. Burden and Faires (1997)). The discrete system is given by

[2]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 the NERC National Centre for Earth Observation website at http://www.nceo.ac.uk/training.php.

$$x_{k+1} = x_k + \sigma\Delta t/2\left[2(y_k - x_k) + \Delta t(\rho x_k - y_k - x_k z_k) - \sigma\Delta t(y_k - x_k)\right], \tag{5.42}$$

$$\begin{aligned} y_{k+1} &= y_k + \Delta t/2\left[\rho x_k - y_k - x_k z_k + \rho(x_k + \sigma\Delta t(y_k - x_k)) - y_k - \Delta t(\rho x_k - y_k - x_k z_k)\right. \\ &- \left.(x_k + \sigma\Delta t(y_k - x_k))(z_k + \Delta t(x_k y_k - \beta z_k))\right] \end{aligned} \tag{5.43}$$

$$\begin{aligned} z_{k+1} &= z_k + \Delta t/2\left[x_k y_k - \beta z_k + (x_k + \Delta t\sigma(y_k - x_k))(y_k + \Delta t(\rho x_k - y_k - x_k z_k))\right. \\ &- \left.\beta(z_k + \Delta t(x_k y_k - \beta z_k))\right] \end{aligned} \tag{5.44}$$

where $\Delta t = 0.01$ is the model time step.

Proceeding as in the previous section we set

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

Our augmented system model is then given by

$$\begin{aligned} \mathbf{w}_{k+1} &= \tilde{\mathbf{f}}(\mathbf{w}_k) \\ &= \begin{pmatrix} \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{x}_k \\ \mathbf{p}_k \end{pmatrix}. \end{aligned} \tag{5.45}$$

where $\mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)$ is the state evolution model given by (5.42)-(5.44) evaluated at $\mathbf{p}_k$.

5.3.1 State-parameter cross covariance

For the Lorenz model we have

$$\begin{aligned} \mathbf{N}_k &= \left(\frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \sigma_k} \quad \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \rho_k} \quad \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \beta_k} \right)\Bigg|_{\mathbf{x}_k^a, \mathbf{p}_k^a} \\ &= \left(\begin{array}{ccc} \frac{\partial x_{k+1}}{\partial \sigma_k} & \frac{\partial x_{k+1}}{\partial \rho_k} & \frac{\partial x_{k+1}}{\partial \beta_k} \\ \frac{\partial y_{k+1}}{\partial \sigma_k} & \frac{\partial y_{k+1}}{\partial \rho_k} & \frac{\partial y_{k+1}}{\partial \beta_k} \\ \frac{\partial z_{k+1}}{\partial \sigma_k} & \frac{\partial z_{k+1}}{\partial \rho_k} & \frac{\partial z_{k+1}}{\partial \beta_k} \end{array} \right)\Bigg|_{\mathbf{x}_{k+1}^b, \mathbf{p}_k^a} \end{aligned} \tag{5.46}$$

Using (5.42), (5.43) and (5.44) we calculate the elements of $\mathbf{N}_k$ as follows

$$\begin{aligned} \frac{\partial x_{k+1}}{\partial \sigma_k} &= \Delta t/2\left[2(y_k - x_k) + \Delta t(\rho_k x_k - y_k - x_k z_k) - 2\sigma_k \Delta t(y_k - x_k)\right] \\ &= \Delta t(1 - \sigma_k \Delta t)(y_k - x_k) + \Delta t^2/2(\rho_k x_k - y_k - x_k z_k), \end{aligned} \tag{5.47}$$

$$\frac{\partial x_{k+1}}{\partial \rho_k} = \sigma_k(\Delta t^2/2)x_k \tag{5.48}$$

$$\frac{\partial x_{k+1}}{\partial \beta_k} = 0 \tag{5.49}$$

$$
\begin{aligned}
\frac{\partial y_{k+1}}{\partial \sigma_k} &= \Delta t/2\left[\rho_k \Delta t(y_k - x_k) - \Delta t(y_k - x_k)(z_k + \Delta t(x_k y_k - \beta_k z_k))\right], \\
&= \Delta t^2/2(y_k - x_k)\left[\rho_k - z_k - \Delta t(x_k y_k - \beta_k z_k)\right] && (5.50) \\
\frac{\partial y_{k+1}}{\partial \rho_k} &= \Delta t/2\left[x_k + x_k + \sigma_k \Delta t(y_k - x_k) - \Delta t x_k\right] \\
&= \Delta t/2\left[(2 - \Delta t)x_k + \sigma_k \Delta t(y_k - x_k)\right] \\
&= (\Delta t - \Delta t^2/2)x_k - \sigma_k \Delta t^2/2(y_k - x_k), && (5.51) \\
\frac{\partial y_{k+1}}{\partial \beta_k} &= \Delta t/2\left[x_k \Delta t z_k + \sigma_k \Delta t(y_k - x_k)\Delta t z_k\right] \\
&= \Delta t^2/2\left[x_k + \sigma_k \Delta t(y_k - x_k)\right] z_k, && (5.52)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial z_{k+1}}{\partial \sigma_k} &= \Delta t/2\left[\Delta t(y_k - x_k)(y_k + \Delta t(\rho_k x_k - y_k - x_k z_k))\right] \\
&= \Delta t^2/2\left[(y_k - x_k)y_k + \Delta t(y_k - x_k)(\rho_k x_k - y_k - x_k z_k))\right] && (5.53) \\
\frac{\partial z_{k+1}}{\partial \rho_k} &= \Delta t/2\left[(x_k + \Delta t \sigma_k(y_k - x_k))\Delta t x_k\right] \\
&= \Delta t^2/2\left[x_k + \sigma_k \Delta t(y_k - x_k)\right] x_k, && (5.54) \\
\frac{\partial z_{k+1}}{\partial \beta_k} &= \Delta t/2\left[-z_k - \Delta t x_k y_k + 2\beta_k \Delta t z_k\right] \\
&= \Delta t^2/2(x_k y_k) + \Delta t/2(2\beta_k \Delta t - 1)z_k && (5.55)
\end{aligned}
$$

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

$$
\mathbf{B_{pp}} = \begin{pmatrix} \sigma_\sigma^2 & 0 & 0 \\ 0 & \sigma_\rho^2 & 0 \\ 0 & 0 & \sigma_\beta^2 \end{pmatrix}, \tag{5.56}
$$

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

The state-parameter cross covariance matrix is then given by

$$
\begin{aligned}
\mathbf{B_{xp}}_{k+1} &= \mathbf{N}_k \mathbf{B_{pp}} \\
&= \left(\sigma_\sigma^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \sigma_k} \quad \sigma_\rho^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \rho_k} \quad \sigma_\beta^2 \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{p}_k)}{\partial \beta_k} \right)\Big|_{\mathbf{x}_k^a, \mathbf{p}_k^a}. && (5.57)
\end{aligned}
$$

5.3.2 Assimilation experiments

We define the 'true' solution to be given by the discrete equations (5.42) - (5.44) with initial conditions $x_0 = -5.4458$, $y_0 = -5.4841$ and $z_0 = 22.5606$. Observation of x, y and z are taken from this solution and assimilated at regular time intervals. The solutions for x and z are shown in figure 5.8. The initial background guess $\mathbf{x}_0^b$ is equal to the true initial conditions 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

$$
\mathbf{B_{xx}} = \sigma_b^2 \mathbf{I}, \qquad \mathbf{I} \in \mathbb{R}^{3\times 3}. \tag{5.58}
$$

The initial estimates of the parameters σ_0, ρ_0 and β_0 are generated by adding random noise with variance equal to 20% of the true value to each of σ, ρ and β. Although by adding the parameters to the state vector we double the dimension of the system, $\mathbf{w} \in \mathbb{R}^{n+q=6}$ is still relatively small and so we use equation (3.3) to compute the analysis directly.

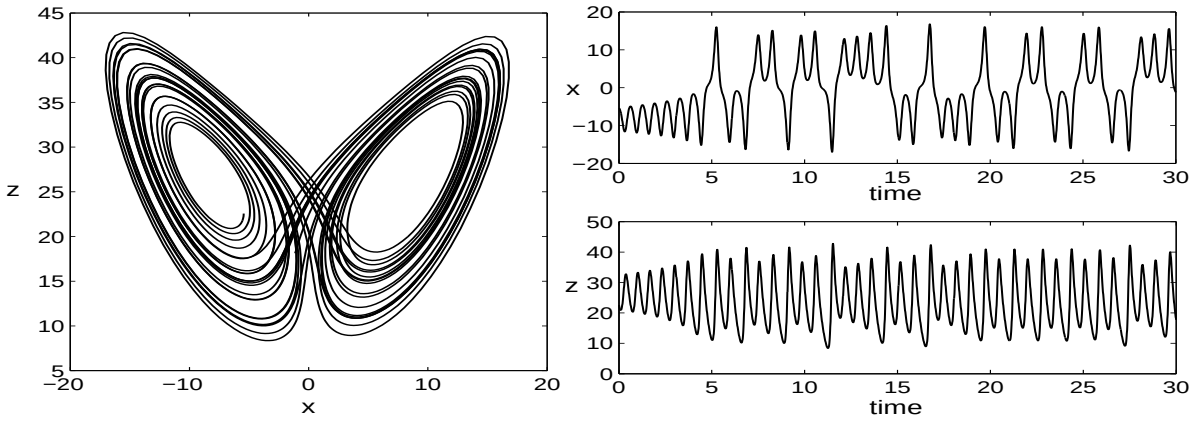

Figure 5.8: Lorenz equations: true solution for x and z.

5.3.3 Results

Perfect observations

Figures 5.9(a) show the parameter estimates produced using the randomly generated starting values $\sigma_0 = 11.0311$, $\rho_0 = 30.1316$ and $\beta_0 = 1.6986$ for observation frequencies $5\Delta t$, $10\Delta t$ and $20\Delta t$, with $\sigma_o^2 = 0.01$. The estimates of ρ and β get close to their true values fairly quickly; the updating of σ is a lot slower but it nonetheless converges to the correct value. Figures 5.9(b) shows the estimates for a observation interval of $30\Delta t$. Here the scheme takes much longer to stabilise and we see much bigger deviations in the estimates. If we try and further increase the period between assimilations to $40\Delta t$ the scheme fails to find the correct parameter values.

Noisy observations

Again, we re-ran our experiments using noisy observations. We used observation error variances $\sigma_o^2 = 0.01$, $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ and assimilated observations at varying time intervals. Figures 5.10 and 5.11 show the parameter updates obtained when the observation error variances were set at 0.1 and 0.25 respectively for initial parameter estimates $\sigma_0 = 9.2405$, $\rho_0 = 25.5385$ and $\beta_0 = 3.395$. The results for $\sigma_o^2 = 0.01$ are not shown as the convergence and quality of the parameter estimates was very similar to the perfect observation case. When we set $\sigma_o^2 = 0.1$ and $\sigma_o^2 = 0.25$ the parameter estimates are extremely noisy; the size of the parameter errors increases as σ_o^2 increases (note the difference in the y axis scale in figure 5.11) and also as the frequency of the observations decreases. We have only shown results for intervals of $5\Delta t$ and $10\Delta t$ as beyond this the oscillations are extremely large. The right hand panels of figures 5.10 and 5.11 show the corresponding time averaged estimates, produced using a moving time window with averaging starting at $t = 1.5$. For assimilation intervals of $5\Delta t$ and $10\Delta t$ the averaged parameter estimates are extremely good. When $\sigma_o^2 = 0.1$ we are also able to get reasonable estimates for an observation interval of $20\Delta t$ by increasing the length of the averaging window, but when $\sigma_o^2 = 0.25$ even the averaged estimates are unreliable.

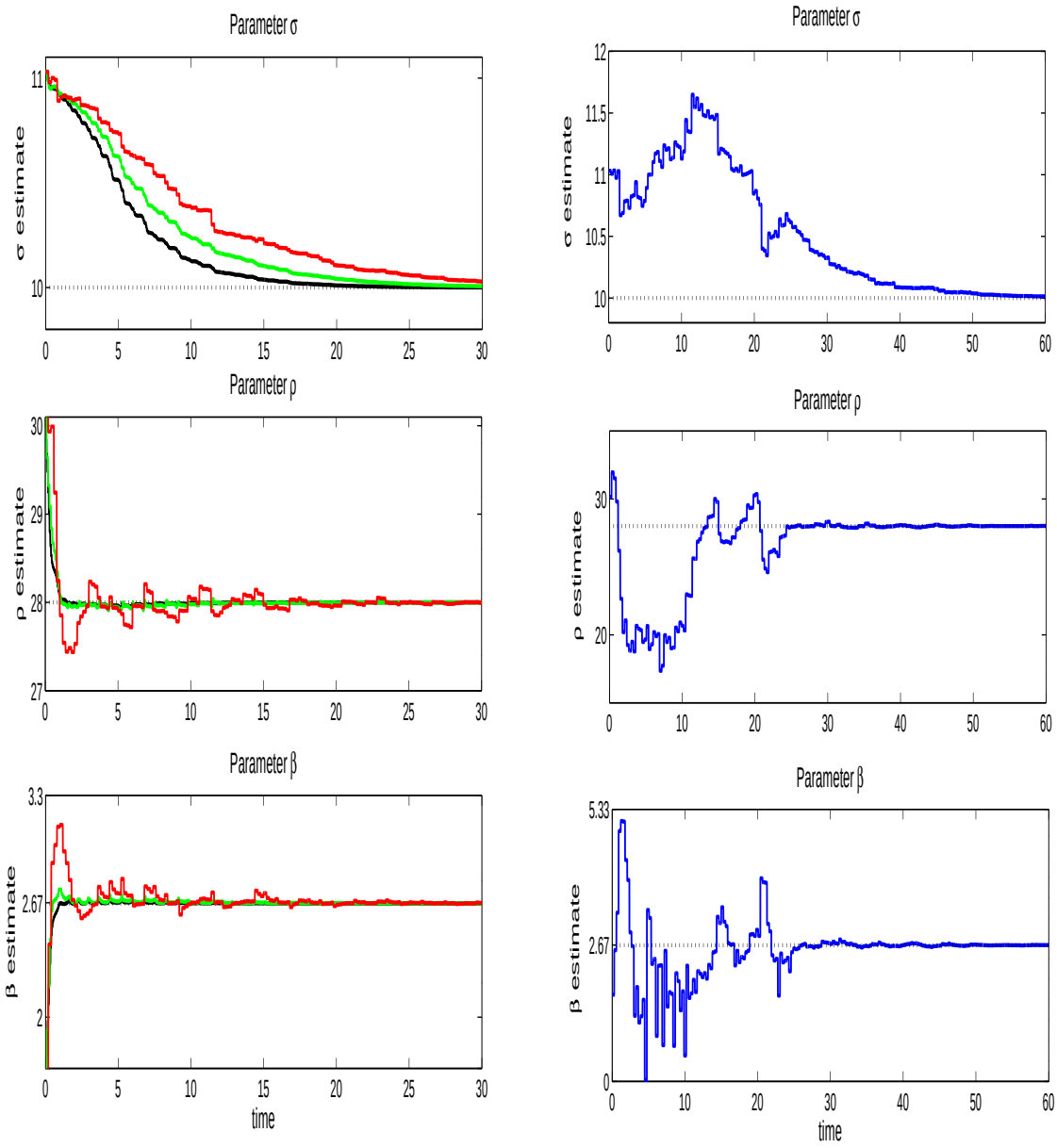

Figure 5.9: **Perfect observations** Parameter updates for initial estimates $\sigma_0 = 11.0311$, $\rho_0 = 30.1316$, $\beta_0 = 1.6986$ **(a)** solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$; solid red line - observations every $20\Delta t$ **(b)** solid blue line - observations every $30\Delta t$.

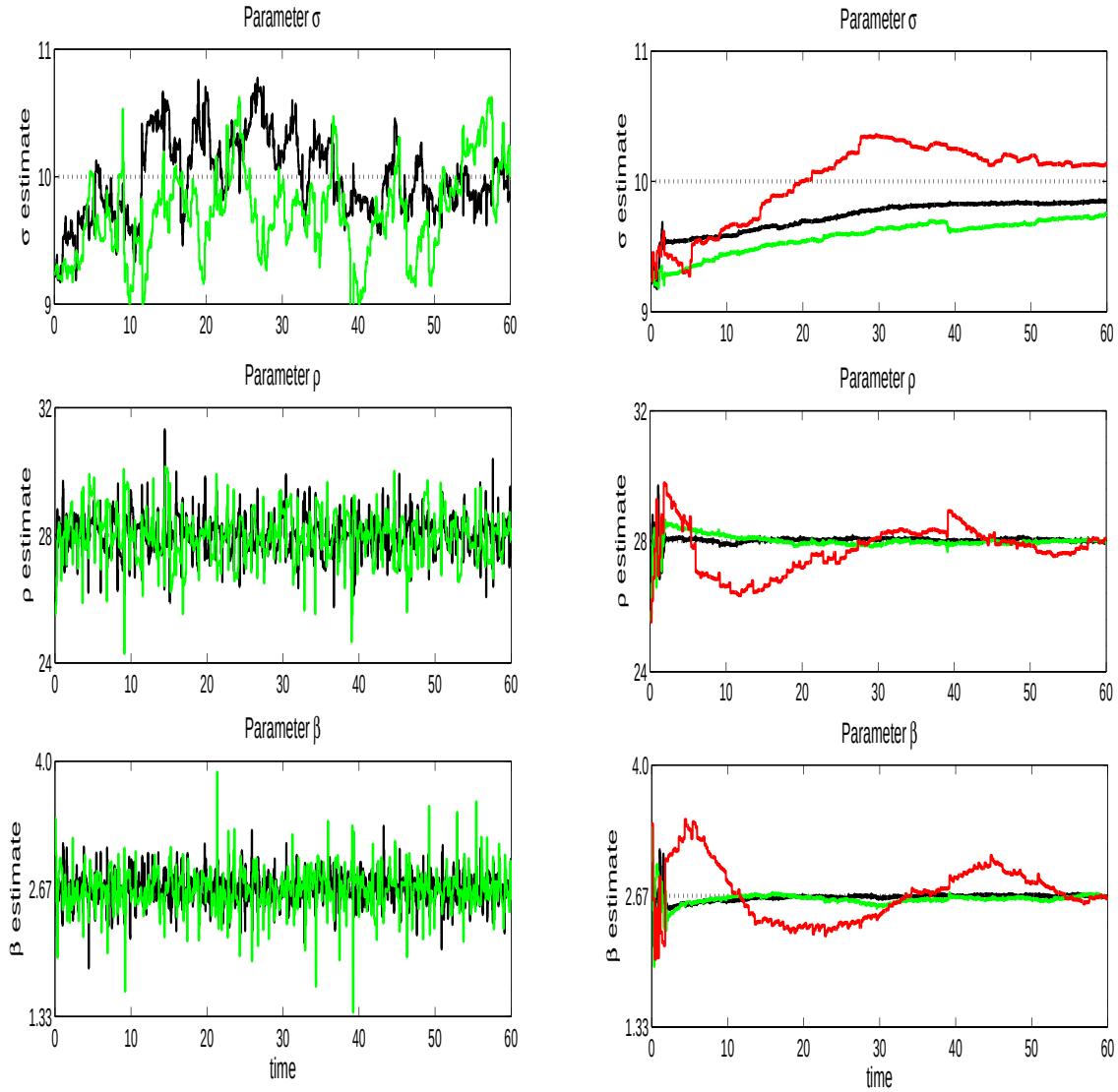

Figure 5.10: **Imperfect observations** Parameter updates for initial estimates $\sigma_0 = 9.2405$, $\rho_0 = 25.5385$ and $\beta_0 = 3.395$, with observation error variance $\sigma_o^2 = 0.1$ **(a)** unaveraged updates, **(b)** averaged updates: solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$

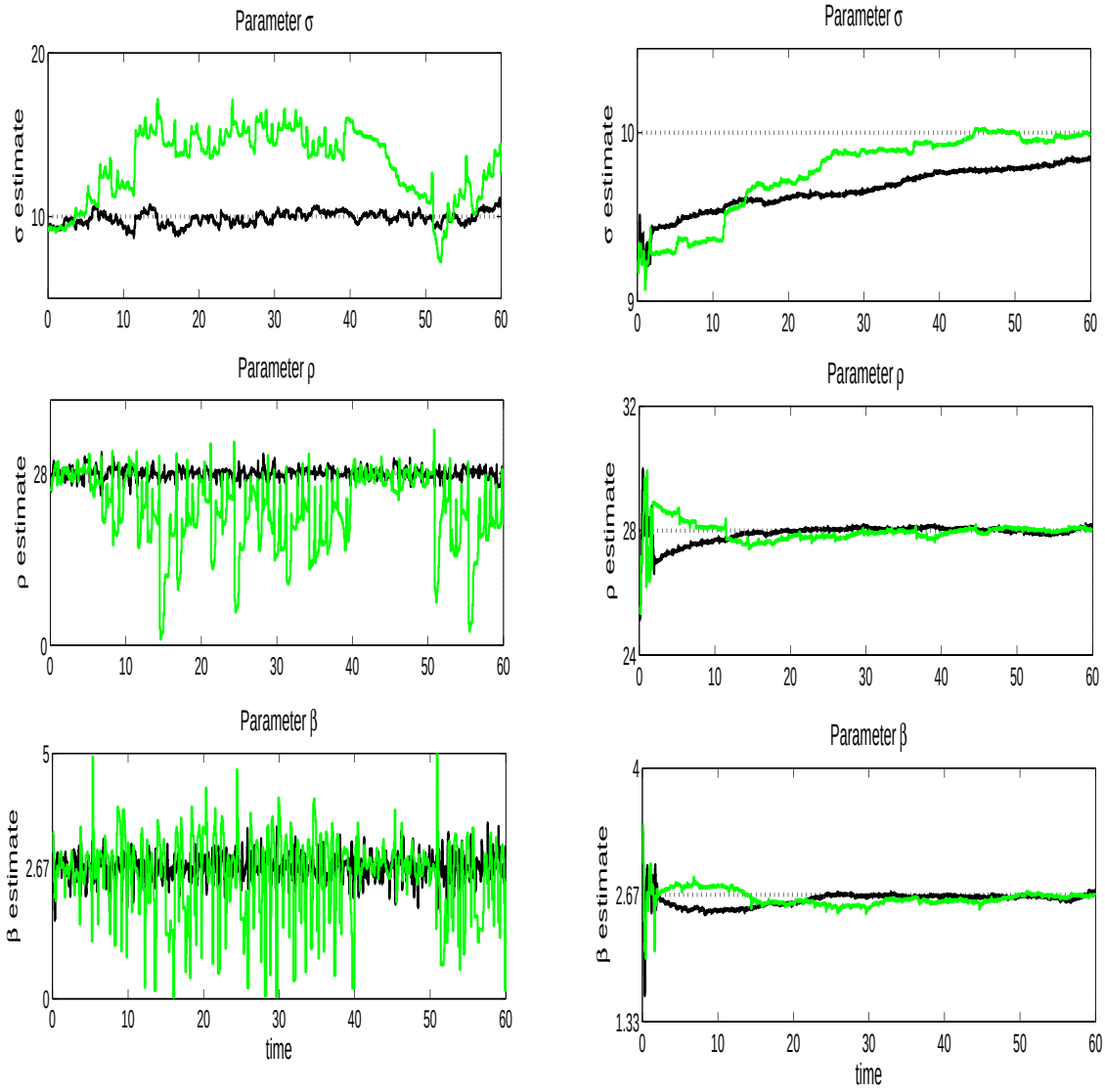

Figure 5.11: **Imperfect observations** Parameter updates for initial estimates $\sigma_0 = 9.2405$, $\rho_0 = 25.5385$ and $\beta_0 = 3.395$, with observation error variance $\sigma_o^2 = 0.25$ **(a)** unaveraged updates, **(b)** averaged updates: solid black line - observations every $5\Delta t$; solid green line - observations every $10\Delta t$.

6 Conclusions

A new method for concurrent model parameter and state estimation, employing a hybrid data assimilation scheme, has been proposed and demonstrated via a series of identical twin experiments with three simple numerical models.

A key difficulty in the construction of a data assimilation algorithm is specification of the background error covariances. These covariances play an important role in the filtering and spreading of observational data and have a direct influence on the quality of the analysis. For parameter estimation, it is the joint state-parameter cross covariances, given by the off-diagonal elements of the augmented state background error covariance matrix $\mathbf{B}$, 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 estimation.

Earlier work (Smith et al. (2008), Smith et al. (2009a)) suggested that the state-parameter covariances should be time-dependent. However, explicitly propagating the $\mathbf{B}$ matrix is computationally expensive and requires the construction of adjoint and tangent linear models. Since the requirement for time dependence did not extend to the estimation of the state error covariances an alternative approach was sought. By combining the ideas from the 3D Var and Kalman filter techniques we have developed a new hybrid scheme that provides a flow dependent approximation of the state-parameter cross-covariances but which avoids the computational complexities associated with implementation of the full Kalman filter equations. This allows us to use the state augmentation technique with 3D Var and OI type algorithms which traditionally make the assumption of a static $\mathbf{B}$.

In this paper we have presented details of this new methodology and illustrated its versatility by applying it to a range of simple dynamical system models in which the use of incorrect parameters has a direct impact on the model solution. As the results show, the scheme performed well in all of the three cases considered and was successful in recovering the parameter values we had specified to a good level of accuracy, even when the observational data were noisy. This had a positive impact on the skill of the forecast model and enabled more accurate predicitions of the true model state.

The method is less successful in situations where the model is relatively insensitive to a particular parameter, as was the case for certain settings in oscillating system. This is not surprising; 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. This raises the issues of observability and identifiability (Barnett and Cameron (1990), Navon (1997)); whether the available observations contain sufficient information for us to be able to determine the parameters of interest and whether these parameters have a unique deterministic set of values. A parameter estimation method can only be expected to work reliably when both these properties hold. Future work will investigate observability and identifiability and how they relate to this new method.

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 (Smith et al. (2009b)). In some situations, it may be prudent to consider a re-parameterisation of the model equations to improve the identifiability of the parameters or even to transform the parameters to a set of uncorrelated variables (Sorooshian and Gupta (1995)).

To date, our new technique has only been tested in models of relatively low dimension, where the number of parameters is small and, since the required parameters are constants, the dynamics of the parameter model are simple. 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 matrix $\mathbf{N}_k$ at each new assimilation time is not infeasible. We note that, although available for the models we used here, a computational tangent linear model is not necessarily required. The matrix $\mathbf{N}_k$ could, for example, be computed using a finite difference approach as described in Smith et al. (2009b). 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.

This study has provided a valuable insight into how the method is likely to perform in a range of dynamical systems. The results are extremely encouraging; the scheme has proved effective in both linear

and non-linear systems. We believe that there is potential for successful application of this new methodology to larger, more realistic models with more complex parameterizations.

Acknowledgements This work is funded under the UK Natural Environmental Research Council (NERC) Flood Risk From Extreme Events (FREE) programme, with additional funding provided by the Environment Agency as part of the CASE (Co-operative Awards in Science and Engineering) scheme.

A Tangent Linear Model (TLM)

Definition

If $\mathbf{f}$ is a non linear model defined as

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

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

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

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 (2.6)

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

We define a perturbation to this state as

$$\boldsymbol{\delta}\mathbf{w}_{k+1} = \mathbf{w}_{k+1} - \widehat{\mathbf{w}}_{k+1}. \tag{A.2}$$

This perturbation then satisfies

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

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

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

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. small $\|\mathbf{w} - \widehat{\mathbf{w}}_k\|_2$.

References

Barnett, S. and Cameron, R. (1990). *Introduction to Mathematical Control Theory*. Oxford Applied Mathematics and Computing Science Series. Oxford Clarendon Press, second edition.

Bell, M., Martin, M., and Nichols, N. (2004). Assimilation of data into an ocean model with systematic errors near the equator. *Quarterly Journal of the Royal Meteorological Society*, 130:873–894.

Burden, R. L. and Faires, D. J. (1997). *Numerical Analysis*. Brooks/ Cole Publishing Company, 6th edition.

Courtier, P., Andersson, E., Heckley, W., Pailleux, J., Vasiljevic, D., Hamrud, M., Hollingsworth, A., Rabier, F., and Fisher, M. (1998). The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Quarterly Journal of the Royal Meteorological Society*, 124:1783–1807.

Dee, D. P. (2005). Bias and data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 131:3323–2243.

Duffing, G. (1918). Erzwungene schwingugen bei veränderlicher eigenfrequenz. *F. Vieweg u. Sohn: Braunschweig*.

Evensen, G. (1994). 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.

Evensen, G., Dee, D. P., and Schröter, J. (1998). Parameter estimation in dynamical models. In Chassignet, E. and Verron, J., editors, *Ocean Modeling and Parameterization*, pages 373–398. Kluwer Academic Publishers.

Fisher, M. (1998). Development of a simplified Kalman filter. Technical Memorandum No. 260, ECMWF Research Department.

Gelb, A. (1974). *Applied Optimal Estimation*. M.I.T Press.

Gill, P. E., Murray, W., and Wright, M. H. (1981). *Practical Optimization*. Academic Press.

Griffith, A. K. and Nichols, N. K. (1996). Accounting for model error in data assimilation using adjoint methods. In Berz, M., Bischof, C., Corliss, G., and Griewank, A., editors, *Computational Differentiation: Techniques, Applications and Tools*, pages 195–204. SIAM Philadelphia.

Griffith, A. K. and Nichols, N. K. (2000). Adjoint techniques in data assimilation for treating systematic model error. *Journal of Flow, Turbulence and Combustion*, 65:469–488.

Guckenheimer, J. and Holmes, P. (1986). *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Applied Mathematical Sciences 42. Springer-Verlag.

Hill, D., Jones, S., and Prandle, D. (2003). Derivation of sediment resuspension rates from accoustic backscatter time-series in tidal waters. *Continental Shelf Research*, 23:19–40.

Houtekamer, P. and Mitchell, H. L. (2005). Ensemble Kalman filtering. *Quarterly Journal of the Royal Meteorological Society*, 131:3269–3289.

Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering (Series D)*, 82:35–45.

Kalman, R. and Bucy, R. (1961). New results in linear filtering and prediction theory. *Transactions of the ASME - Journal of Basic Engineering (Series D)*, 83:95–108.

Knaapen, M. and Hulscher, S. (2003). Use of genetic algorithm to improve prediction of alternate bar dynamics. *Water Resources Research*, 39(9):1231.

LeVeque, R. J. (1992). *Numerical Methods for Conservation Laws*. Birkhäuser-Verlag.

Lewis, J. M., Lakshmivarahan, S., and Dhall, S. (2006). *Dynamic Data Assimilation: A Least Squares Approach*, volume 104 of *Encyclopedia of Mathematics and its applications*. Cambridge University Press.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20:130–141.

Martin, M., Bell, M., and Nichols, N. (2002). Estimation of systematic error in an equatorial ocean model using data assimilation. *International Journal for Numerical Methods in Fluids*, 40:435–444.

Martin, M., Nichols, N., and Bell, M. (1999). Treatment of systematic errors in sequential data assimilation. Technical Note No. 21, Meteorological Office, Ocean Applications Division.

Martin, M. J. (2000). *Data Assimilation in Ocean Circulation Models with Systematic Errors*. PhD thesis, University of Reading. Available at http://www.reading.ac.uk/maths/research/.

Navon, I. M. (1997). Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. *Dynamics of Atmosphere and Oceans*, 27:55–79.

Rodgers, C. D. (2000). *Inverse Methods for Atmospheric Sounding: Theory and Practice*, volume 2 of *Series on Atmospheric, Oceanic and Planetary Physics*. World Scientific.

Ruessink, B. (2005a). Calibration of nearshore process models - application of a hybrid genetic algorithm. *Journal of Hydroinformatics*, 7:135–149.

Ruessink, B. (2005b). Predictive uncertainity of a nearshore bed evolution model. *Continental Shelf Research*, 25:1053–1069.

Ruessink, B. (2006). A Bayesian estimation of parameter-induced uncertainty in a nearshore alongshore current model. *Journal of Hydroinformatics*, 7:37–49.

Smith, P. J., Baines, M. J., Dance, S. L., Nichols, N. K., and Scott, T. R. (2008). Data assimilation for parameter estimation with application to a simple morphodynamic model. Mathematics Report 2/2008, Department of Mathematics, University of Reading. Available at http://www.reading.ac.uk/maths/research/.

Smith, P. J., Baines, M. J., Dance, S. L., Nichols, N. K., and Scott, T. R. (2009a). Variational data assimilation for parameter estimation: application to a simple morphodynamic model. *Ocean Dynamics*, 59(5):697–708.

Smith, P. J., Dance, S. L., and Nichols, N. K. (2009b). Data assimilation for morphodynamic model parameter estimation: a hybrid approach. Mathematics Report 2/2009, Department of Mathematics, University of Reading.

Sorooshian, S. and Gupta, V. (1995). Model calibration. In Singh, V., editor, *Computer models of watershed hydrology*, chapter 2, pages 23–68. Water Resources Publications, Colorado.

Sparrow, C. (1982). *The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors*. Applied Mathematical Sciences 41. Springer-Verlag.

Stoker, J. (1966). *Nonlinear Vibrations in Mechanical and Electrical Systems*. Pure and Applied Mathematics Volume II. Interscience.

Thompson, J. and Stewart, H. (1986). *Nonlinear Dynamics and Chaos: Geometrical Methods for Engineers and Scientists*. John Wiley and Sons Ltd.

Trudinger, C., Raupach, M., Rayner, P., and Enting, I. (2008). Using the Kalman filter for parameter estimation in biogeochemical models. *Environmetrics*, 19(8):849–870.

Wiggins, S. (1990). *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Texts in Applied Mathematics 2. Springer-Verlag.

Wüst, J. (2004). Data-driven probablistic predictions of sand wave bathymetry. In S.J.M.H., H., Garlan, T., and Idier, D., editors, *Marine Sandwave and River Dune Dynamics II*. Proceedings of International Workshop, University of Twente.

Zupanski, D. and Zupanski, M. (2006). Model error estimation employing an ensemble data assimilation approach. *Monthly Weather Review*, 134.