THE UNIVERSITY OF READING

DEPARTMENTS OF MATHEMATICS AND METEOROLOGY

4D-Var for high resolution, nested models with a range of scales

Gillian M. Baxter

Thesis submitted for the degree of

Doctor of Philosophy

June 2009

Abstract

Hazardous or extreme weather is often caused by localised, convective scale features. The high social and economic impact of such events has created a growing need within numerical weather prediction to improve the accuracy of convective scale forecasting. Many operational centres around the world are developing forecasting models with spatial resolutions down to 1km. Such high resolution allows the dominant motions at the convective scale to be generally well represented.

Initial conditions for numerical forecasting models are provided using data assimilation, a method which combines observations with a current estimate of the atmospheric state. However, due to limits in computer power, very high resolution models can only cover a limited area domain. This means that there may be phenomena present in the atmosphere with length scales that are longer than the domain of the model.

It is important that the data assimilation on the limited area model (LAM) capture both the small and large scales, including lengthscales longer than the LAM domain. To investigate how different scales are treated in a LAM data assimilation, we nest a limited area domain within a 1D linear advection-diffusion model and implement a four dimensional variational (4D-Var) data assimilation scheme. We use a discrete Fourier sine transform to investigate which wavelengths are captured by the data assimilation.

We show that, with high resolution observations, the LAM data assimilation is able to accurately represent the small scales. However, it cannot capture the truth exactly, due to errors coming from the boundary conditions. We demonstrate that errors from the boundary conditions, as well as limitations in the resolution of the LAM, cause errors at low wavenumbers. We also show that lengthscales longer than the LAM domain are aliased onto other wavenumbers, with the majority of the information being aliased onto the longest waves contained by the LAM spectrum. Using this knowledge we develop a new method for improving the low wavenumbers within the LAM 4D-Var whilst still maintaining the accuracy in the small scales achieved by the high resolution.

Declaration

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

Gillian Baxter

Acknowledgements

Firstly I would like to thank my academic supervisors, Dr Sarah Dance, Dr Amos Lawless and Prof. Nancy Nichols, for their help, guidance, support and patience. I would also like to thank my Met Office supervisor, Sue Ballard.

My thanks go to the staff and students of the Mathematics and Meteorology departments, with whom I have had many useful discussions. I particularly thank the members of the student data assimilation group and my monitoring committee members: Dr Peter Sweby, Dr David Stephenson and Dr Robert Plant.

I thank all of my friends in the Mathematics department and elsewhere, for their friendship and support over the past four years. I am particularly grateful to Jenny, all the girls who have shared Office 300 and all my housemates. Your friendship, moral support, and in the case of Jenny; your extensive Fortran knowledge and endless patience, have been invalauble.

I also wish to thank my family; for the love and encouragement you always give me and for keeping me sane (well, at least as sane as I ever was!) during the past four years.

Finally, I acknowledge the financial support of the Natural Environment Research Council (NERC) and the UK Met Office.

Contents

List of Figures

Acronyms

NWP	Numerical weather prediction
LAM	Limited area model
LBC	Lateral boundary condition
3D-Var	3 dimensional variational data assimilation
4D-Var	4 dimensional variational data assimilation
KF	Kalman filter
EnKF	Ensemble Kalman filter
DFT	Discrete Fourier transform
FFT	Fast Fourier transform

Data Assimilation Notation

$\boldsymbol{x}$	state vector
$\boldsymbol{x^a}$	analysis vector
$\boldsymbol{x^t}$	'true state' vector
$\boldsymbol{x^b}$	background state/prior estimate
$\boldsymbol{y}$	observation vector
$\boldsymbol{h}$	observation operator
$\boldsymbol{H}$	linearisation of observation operator $\boldsymbol{h}$
$\boldsymbol{B}$	background error covariance matrix
$\boldsymbol{\Sigma}$	background error covariance matrix in spectral space
$\boldsymbol{R}$	observation error covariance matrix
$\boldsymbol{m}$	non-linear model
$\boldsymbol{M}$	linear forward model
$\boldsymbol{M^T}$	adjoint model
$\boldsymbol{K}$	gain matrix
J	cost function
J_b	background term of cost function
J_o	observation term of cost function

Model Notation

u	Temperature variable
x^P	Parent spatial coordinate
x^L	LAM spatial coordinate
t	Time
c	Advection velocity
σ	Diffusion constant
Δx^P	Parent grid spacing
Δt^P	Parent timestep
Δx^L	LAM grid spacing
Δt^L	LAM timestep
N	Number of parent gridpoints
M	Number of LAM gridpoints
$\mathcal{T}$	Number of parent model timesteps in the assimilation window
S	Number of LAM timesteps in the assimilation window
h	Ratio of LAM to parent gridspace
τ	Ratio of LAM to parent timestep
D	Number of parent model gridspaces covered by LAM
$B1$	Parent gridpoint corresponding to first LAM boundary
$B2$	Parent gridpoint corresponding to second LAM boundary
b	Width of buffer zone

Chapter 1

Introduction

The ultimate problem in meteorology [18, p. 6].

This bold statement has been used to describe the problem of forecasting the weather [11, Ch. 1], [18, p. 6]. According to Bjerknes, to produce the best weather forecast it is necessary to satisfy two conditions, which define weather prediction as an initial-value problem [18, p. 7]:

1. *The present state of the atmosphere must be characterised as accurately as possible.*
2. *The intrinsic laws, according to which the subsequent states develop out of the preceding ones, must be known.*

The first condition is addressed in current numerical weather prediction (NWP) systems using a method known as data assimilation. Data assimilation is the process of combining a previous model forecast (background) with observations to generate initial conditions that most accurately describe the observed reality. The second condition is addressed within a numerical model. The aim is to have a model which satisfies the known physical properties of the atmosphere as accurately as possible, given the restrictions imposed by computer power, our ability to solve the equations numerically, and our ability to understand fully the processes governing the atmosphere.

1.1 Motivation

When forecasting the weather there are many scales to consider. We might be forecasting synoptic scale phenomena, of order 1000km or more. Most of the high and low pressure areas seen on weather maps are on the synoptic scale. Alternatively, we might be forecasting more localised phenomena that occur on the mesoscale or convective scale (of order a few to several hundred km). Squall lines, fronts and storm bands are all mesoscale weather events. Individual thunderstorms occur on the convective scale.

Hazardous or extreme weather is often caused by convective scale features. Convective storms produce some of the most damaging weather experienced in the UK [60]. The leading cause of floods in the UK over the summer months is severe rainfall from convective events [36]. For example, it was a convective storm that led to the flood in Boscastle, UK, in 2004 [32]. The high social and economic impact of such floods has created a growing need within numerical weather prediction (NWP) to improve our ability to forecast these storms accurately.

A reason for the difficulty in predicting convective storms is that in order to represent them accurately, spatial resolutions of order 100m are needed to resolve the dominant motions properly [10]. However, running operational forecast models at such high resolutions would require computer power beyond that currently available. Nevertheless, research at the UK Met Office has shown that very useful results can be obtained with horizontal resolutions of order 1km, because the mesoscale flows leading to storm initiation, organisation and propagation are generally well represented [51], [75]. The 1km resolution also allows us to do away with convective parameterisations and to use a more accurate orography, for example, at the coastal boundary. Coastal effects can be important triggering mechanisms on the convective scale [9]. Sea-breezes, which are triggered by the land-sea boundary, played a key role in the initiation of the storm and subsequent flash flooding in Boscastle [32].

Despite technological improvements there are still limits to computer power however, and if a model is to run at a very high resolution it can only have a limited domain size. For example, the UK Met Office runs a 4km resolution model centred over the UK, with a horizontal domain of 288×360 grid points (1152km×1440km) and 70 vertical levels [61]. This domain is currently being upgraded to 1.5km resolution.

The limited area nature of high resolution models can cause problems for a data assimilation algorithm. One problem is the need for lateral boundary conditions (LBCs) and the errors these LBCs introduce. Another problem is the range of scales a limited area model (LAM) is able to capture. It is important that the LAM be able to capture the small scale detail in order to accurately forecast convective features. However, due to the limited area domain there may be phenomena present in the atmosphere with length scales longer than the domain of the model and it is also important that the LAM be able to accurately represent these larger scales.

1.2 Goals of thesis

The aims of this thesis are:

- To consider some of the problems caused by the limited area nature of high resolution models.
- To investigate the treatment of different length scales by 4D-Var (four dimensional variational data assimilation) on a LAM domain. We consider both small and large scales, including lengthscales longer than the domain of the LAM.
- To use the knowledge gained about the treatment of different length scales to demonstrate a possible method of improving a LAM 4D-Var solution.

1.3 Principal results of thesis

The principal results that are shown in this thesis are:

1. Given high resolution observations, the LAM analysis is able to accurately represent the small scales.

2. Even with perfect observations and no background term, the LAM data assimilation cannot capture the truth exactly, due to errors introduced through the LBCs.

3. Errors introduced through the LBCs, as well as differences in resolution between the truth and the LAM, cause the LAM data assimilation to have errors at low wavenumbers.

4. In the LAM data assimilation waves with lengthscales longer than the domain of the LAM are aliased onto other wavenumbers, with the majority of the information being aliased onto the longest waves contained by the LAM spectrum.

5. In spectral space, the background error covariance matrix $\mathbf{\Sigma}$ can be used to control how the different wavenumbers are affected by the data assimilation.

6. The matrix $\mathbf{\Sigma}$ can be specified in such a way as to improve the representation of the low wavenumbers whilst also allowing the smaller scales to be accurately represented.

1.4 Thesis outline

In Chapter 2 we introduce data assimilation, what it is and why we use it. We briefly introduce some of the many different types of data assimilation before focussing in detail on 4D-Var, which is the type of data assimilation to be used for the work presented in this thesis.

In Chapter 3 we introduce the discrete Fourier transform (DFT) and the power spectrum. We demonstrate features of the power spectrum and how they enable us to use the DFT as a tool to understand the analysis generated by the data assimilation.

Limited area models (LAMs) are described in Chapter 4. We introduce some of the LAMs currently in use in operational weather centres around the world. We also discuss some of the problems the limited area causes for both the model and the data assimilation, in particular, the need for lateral boundary conditions (LBCs) and the problem of having length scales in the atmosphere longer than the domain of the model.

In Chapter 5 we set up a coarse and a fine, nested grid and discuss the adaptations made to the standard 4D-Var scheme to apply it on these domains. We outline a 4D-Var scheme in both physical and spectral space.

In Chapter 6 we investigate the features of the DFT, introduced in Chapter 3, but now applied over a LAM domain.

The model used in this thesis is described in Chapter 7. We outline the design of the experiments and how observations are generated for the data assimilation.

The treatment of both long and short waves by a 4D-Var scheme on a LAM domain are investigated in Chapter 8. In particular, we demonstrate the effects of the lateral boundary conditions on the accuracy of the LAM data assimilation and demonstrate how different scales are treated by the 4D-Var.

In Chapter 9 we apply the results of Chapter 8 to develop a method for manipulating the LAM analysis by controlling which wavenumbers are affected by the 4D-Var.

We conclude in Chapter 10 and discuss possible future work.

Chapter 2

Data Assimilation

To generate a numerical weather forecast we need not only a numerical weather prediction (NWP) model but also a best estimate of the initial values of the atmospheric state [52]. The sources of information we have to provide this estimate are the model itself and any available observations. The initial conditions could be provided by a previous model forecast. However, no model is perfect and there will always be model error. The atmosphere is a continuous system and we are approximating it on a discrete, finite grid. Sub-grid scale effects are parameterised and equations governing physical processes often have to be simplified before we can solve them numerically or have the computer power to do so. Even with a perfect model there is still the difficulty that the atmosphere is a chaotic system [59]. Even tiny errors in the model's initial conditions will magnify over time rendering the forecast useless. We therefore use observations to constrain the system and keep the model output grounded in reality.

Just as the previous model forecast is not enough by itself to provide the initial conditions, neither are the observations sufficient to provide the initial conditions on their own. There are several reasons for this [64]:

1. The observations can and do have errors.
2. The observations may not be direct observations of model variables.
3. The state obtained purely from observations may not satisfy the physical properties and dynamical equations that govern the model.

4. We do not have sufficient observations to give a complete representation of the atmospheric state.

In order to generate initial conditions that accurately describe the observed reality we therefore combine a previous model forecast (background) with observations. The tool that allows us to do this is data assimilation and the initial conditions are known as the 'analysis' [50, p.13], [64].

An NWP model requires an estimate of the state that is accurate at the initial time. However, the aim of a weather prediction model is to produce an accurate forecast. The aim of data assimilation therefore has to be to generate an analysis that most accurately describes the observed reality at the present (initial) time but also one that will generate an accurate estimate of the future state of the system [64].

Many types of data assimilation have been developed, varying in computational cost, optimality and speed [12]. In this chapter we introduce a few of the different types of data assimilation schemes and illustrate the main differences between the varying approaches. We then go on to consider the scheme to be used in the work in this thesis in more detail, demonstrating its derivation and discussing its practical implementation.

2.1 Types of data assimilation

There are many different types of data assimilation schemes. These schemes can be sequential or variational and three-dimensional (space) or four-dimensional (space-time).

Sequential algorithms, such as Optimal Interpolation (OI) [50, Section 5.4.1], [54] or the Kalman filter (KF) [49], use optimal estimation equations to compute the analysis explicitly. Variational algorithms, such as 3D-Var (three-dimensional variational data assimilation) [56], [70], or 4D-Var (four-dimensional variational data assimilation) [16], [50, Section 5.6.3], compute the analysis by minimising a cost function J as a least squares problem.

In three-dimensional schemes, observations are distributed in space only and are taken to be valid at one single time level. In four-dimensional schemes, observations are distributed in time as well as space.

Variational data assimilation is the method of choice in many numerical weather prediction centres and is expected to remain so for the foreseeable future [7]. For this reason we now discuss the variational techniques in more detail.

2.1.1 Notation

Before we discuss the algorithms themselves we begin by defining the notation to be used here. We base this notation on that outlined by [44]. We have:

- a state vector $\boldsymbol{x} \in \mathbb{R}^N$, which describes the column vector of variables that represent the state of the system and defines our model space.
- an analysis vector $\boldsymbol{x^a} \in \mathbb{R}^N$, which is the solution generated by the assimilation scheme.
- a 'true state' $\boldsymbol{x^t} \in \mathbb{R}^N$ valid at the analysis time. This is what we are trying to estimate with the analysis.
- a background state/prior estimate $\boldsymbol{x^b} \in \mathbb{R}^N$, usually provided by an earlier forecast.
- a vector of observations $\boldsymbol{y} \in \mathbb{R}^P$.
- a matrix $\boldsymbol{h}$, and its linearisation (or Jacobian) $\boldsymbol{H}$, where $\boldsymbol{h}$ is an observation operator which is applied to the state vector to convert from model to observation space, $\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x^t}) + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon}$ is the error. For example in the case of a direct observation it might be the interpolation of a wind observation that is not located at a grid point, for an indirect observation it might be the physical relationships connecting an observed radar reflectivity and the model variables. $\boldsymbol{h}$ maps from $\mathbb{R}^N$ to $\mathbb{R}^P$.
- a covariance matrix $\boldsymbol{B} \in \mathbb{R}^{N\text{x}N}$ associated with the background errors $\boldsymbol{\eta}$, where $\boldsymbol{x^b} = \boldsymbol{x^t} + \boldsymbol{\eta}$. The background errors $\boldsymbol{\eta}$ are assumed to be Gaussian and unbiased.
- a covariance matrix $\boldsymbol{R} \in \mathbb{R}^{P\text{x}P}$ associated with the observation errors $\boldsymbol{\epsilon}$. The observation errors $\boldsymbol{\epsilon}$ are assumed to be Gaussian and unbiased.

2.1.2 Variational schemes

In variational assimilation algorithms we find the analysis $\boldsymbol{x^a}$ which minimises a cost function as a least squares problem. The cost function is minimised directly, using an iterative scheme, to find the analysis [56], [64].

A 3D-Var scheme [56], [70] finds the analysis by minimising the cost function

$$J(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x^b})^T \boldsymbol{B}^{-1}(\boldsymbol{x} - \boldsymbol{x^b}) + \frac{1}{2}(\boldsymbol{y} - h(\boldsymbol{x}))^T \boldsymbol{R}^{-1}(\boldsymbol{y} - h(\boldsymbol{x})). \tag{2.1}$$

The cost function J is a measure of the distance of the solution from the background $\boldsymbol{x^b}$ and the observations $\boldsymbol{y}$, weighted by the inverse of the error covariance matrices $\boldsymbol{B}$ and $\boldsymbol{R}$ respectively. In 3D-Var the observations $\boldsymbol{y}$ are all assumed to be valid at a single time and there is no time dependence in the cost function.

4D-Var is closely linked to 3D-Var [71]. In 4D-Var however, a time-sequence of observations can be used by utilising the model dynamics. As with 3D-Var, 4D-Var looks for the analysis as a solution of a minimisation problem, but now using the cost function

$$J(\boldsymbol{x_0}) = \frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{x^b})^T \boldsymbol{B}^{-1}(\boldsymbol{x_0} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k}))^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})), \tag{2.2}$$

subject to the strong constraint that the model states must also be a solution of the model equations

$$\boldsymbol{x_k} = \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}}), \tag{2.3}$$

where k is the timestep, $\mathcal{T}$ is the total number of timesteps in the assimilation window and $\boldsymbol{m}$ is the nonlinear model operator [16], [50, Section 5.6.3], [70].

A strong constraint means that we assume the model equations to be perfect [17]. However, due to the complex nature of the atmosphere our model will never actually be perfect. It is possible to formulate the 4D-Var problem with a weak constraint instead so that the perfect model assumption can be relaxed. This results in an additional term in the cost function to account for model error [17]. In this thesis however, we will assume a perfect model and not include this term in our cost function. As well as an additional term to account for model error, it is also possible to impose other additional weak constraints. One such example for a limited area domain is discussed in Section 4.2.1.

The difference between the 4D-Var cost function (2.2) and the cost function for 3D-Var (2.1) is the time dependence in the observation term. A 4D-Var scheme is run over an assimilation window and the observations are allowed to be distributed in time. The model equations are evolved forward in time and then the values predicted by the model are compared with the observations at the time the observations are valid. We try to find an analysis $\boldsymbol{x^a}$ such that if we run the model forward from $\boldsymbol{x^a}$ we get a best fit trajectory through the observations and the background.

2.1.3 The Kalman Filter

In contrast to the variational technique used by 4D-Var, the Kalman Filter (KF) is a 4-dimensional sequential method [49]. It too allows observations to be assimilated over a time window. However, the KF steps through time, assimilating observations timestep by timestep, to give the optimal analysis at each observation time, given all previous information.

The KF can be viewed in two stages: forecast (denoted f) and analysis (denoted a). The forecast stage comprises of the state forecast

$$\boldsymbol{x_k^f} = \boldsymbol{M_{k-1} x_{k-1}^a}, \tag{2.4}$$

and the error covariance forecast

$$\boldsymbol{P_k^f} = \boldsymbol{M_{k-1} P_{k-1}^a M_{k-1}^T}, \tag{2.5}$$

where $\boldsymbol{M_k}$ is the linear model at time t_k and $\boldsymbol{P_k}$ is the error covariance matrix at time t_k, such that at the beginning of the assimilation window $\boldsymbol{P_0} = \boldsymbol{B}$, where $\boldsymbol{B}$ is the error covariance matrix used in 4D-Var. For simplicity we have neglected the model error term from the error covariance forecast here but it can be included in the formulation. The analysis stage comprises of the Kalman gain computation

$$\boldsymbol{K_k} = \boldsymbol{P_k^f H_k^T} \left[\boldsymbol{H_k P_k^f H_k^T} + \boldsymbol{R_k}\right]^{-1}, \tag{2.6}$$

the state analysis

$$\boldsymbol{x_k^a} = \boldsymbol{x_k^f} + \boldsymbol{K_k} \left[\boldsymbol{y_k} - \boldsymbol{H_k x_k^f}\right], \tag{2.7}$$

and the error covariance of the analysis

$$\boldsymbol{P_k^a} = \left[\boldsymbol{I} - \boldsymbol{K_k H_k}\right] \boldsymbol{P_k^f}, \tag{2.8}$$

where $\boldsymbol{x}_{\boldsymbol{k}}^{\boldsymbol{a}}$ is the analysis at time t_k, with the associated error covariance $\boldsymbol{P}_{\boldsymbol{k}}^{\boldsymbol{a}}$.

In the least squares sense, the KF algorithm is the optimal way to assimilate observations sequentially. It provides the best linear unbiased estimate of the state and its error covariance [50, 5.6.1].

2.1.4 Why 4D-Var is the method of choice

One advantage of variational algorithms is that they can handle indirect observations in their 'raw' format, eliminating the need for retrieval operations [65]. They can also link observations to model variables in a nonlinear manner and project information from model space to observation space, and vice versa, via nonlinear observation operators [19], [65].

4D-Var also has the advantage that observations can be distributed in time as well as space, by utilising the model dynamics. This means that not only are the observations being used at the appropriate time, but also that several observations at the same location can be used within one assimilation period [71]. 4D-Var can extract dynamically consistent information from a time-series of observations [57]. When fitting to the observations, solutions which are inconsistent with the dynamical equations can be penalised [19]. This temporal benefit is not available from 3D-Var as the observations are assumed to be valid at a single time level. It is however, available from the Kalman filter.

The KF algorithm is the optimal way to assimilate observations sequentially (in the least squares sense). It provides the best linear unbiased estimate of the state and its error covariance [50, 5.6.1]. There are similarities between 4D-Var and the KF. Over the same time interval, assuming a linear model, a perfect model, Gaussian errors and that both algorithms use the same data (including the same background information at t_0), the final analysis produced by the KF is equal to the final value of the optimal trajectory estimated by 4D-Var [16], [70]. However, intermediate trajectory values will be different and 4D-Var values will be more accurate since 4D-Var assimilates all the observations simultaneously. It has also been shown that 4D-Var is more successful in reconstructing the physical fields in data void areas [24].

Both 4D-Var and the KF benefit from using evolved error covariances [57]. However, a major advantage of the KF is that it advances the background error covariances via an error covariance forecast step. This error covariance forecast provides a flow dependent background error covariance matrix at the end of the time window, rather than estimating it as a constant covariance matrix as is done in 4D-Var. Although 4D-Var does provide an implicit propagation of the error covariance matrix [65], these propagated covariances are not available as an output at the end of the 4D-Var assimilation. Hence, if we are cycling with 4D-Var, we have to use the same background error covariance matrix at the start of each assimilation window, rather than an updated version as would be provided by the KF. However, NWP models are too large to use a full KF. The error covariance forecast is too computationally expensive due to the error covariance forecast covariance matrix being typically of O(10^7x10^7). The KF does provide a useful reference for the design of approximate algorithms such as the ensemble Kalman filter (EnKF) [42], though due to the approximations made these schemes are no longer optimal in the least squares sense.

A major difficulty with 4D-Var is that to minimise the cost function we use an adjoint model (This is discussed in Section 2.3.1). The adjoint model can be difficult to code and requires rigourous testing.

The benefits provided by 4D-Var make it worthwhile however, and variational data assimilation is the method of choice in many numerical weather prediction centres and is expected to remain so for the foreseeable future [7]. 4D-Var has been operational at ECMWF since November 1997 [71], at Météo-France since 2000 [28] and the UK Met Office since 2004 [57].

4D-Var also has benefits in the context of mesoscale and storm-scale applications, in particular, the ability to use indirect observations and a nonlinear observation operator, as most high resolution data comes from remote sensing and does not measure model variables directly. The data can also have a high temporal, as well as spatial, resolution and 4D-Var can take advantage of this. The status and progress of 4D-Var and in particular the problems associated with applying it to the prediction of storm-scale atmospheric phenomena are reviewed by [19], [58], [65] and [81].

Variational data assimilation is being and will be used with increasingly higher resolution, limited area models. 4D-Var is currently run at JMA (Japan meteorological Agency) at 5km resolution [46]. The UK Met Office 4km model is currently run with 3D-Var with the aim of moving to 4D-Var [22] and there are already plans published by Météo-France for a 3D-Var-based high-resolution limited-area system with a model of 2.5km grid length [26].

Given the benefits of 4D-Var for high resolution applications and that 4D-Var is being aimed for operationally as the method of choice, 4D-Var is the method we use in this thesis.

In the rest of this Chapter we go on to discuss 4D-Var in greater detail, in particular its formulation and practical implementation.

2.2 The mathematical formulation of 4D-Var

In order to understand the 4D-Var scheme we begin by discussing how 4D-Var is formulated.

The 4D-Var problem can be formulated mathematically using linear algebra or by using the technique of calculus of variations. Although in most practical applications the linear algebra approach is used, it is more illuminating to first consider the formulation using calculus of variations. As we are working with a discrete model in this thesis, we consider the discrete formulation here.

For the discrete 4D-Var problem we aim to minimize the cost functional (equation (2.2)),

$$J(\boldsymbol{x_0}) = \frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{x^b})^T \boldsymbol{B}^{-1}(\boldsymbol{x_0} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k}))^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})),$$

subject to the strong constraint (as given by equation (2.3)) that

$$\boldsymbol{x_k} = \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}}),$$

where the subscript k denotes quantities at observation time k from $k = 0$ to $k = \mathcal{T}$ and $\boldsymbol{m}$ is a nonlinear function.

The cost function (2.2) measures the distance between the model state and the observations and between the model state and the background, weighted by the inverse of the error covariance matrices. The strong constraint (2.3) here is the model equation, which describes the dynamics of the discrete system. The model equation is assumed to be error free i.e. we assume we have a perfect model. This assumption is also often made in operational systems.

Following the method of Lagrange multipliers shown in [18, Section 13.2 and Appendix C], considering only the observational term for simplicity, we seek the minimum of

$$J_o(\boldsymbol{x_0}) = \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k}))^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})), \tag{2.9}$$

subject to the constraint (2.3), which can also be written as

$$\boldsymbol{x_k} - \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}}) = 0. \tag{2.10}$$

The problem can be rewritten using discrete Lagrange multipliers which take into account the additional information provided by the constraint (2.10). The whole problem can now be expressed as the minimisation with respect to $\boldsymbol{x_k}$ and $\boldsymbol{\lambda_k}$ of

$$\mathcal{L} = \sum_{k=0}^{\mathcal{T}}\left\{\frac{1}{2}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k}))^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})) + \boldsymbol{\lambda_k}^T\left(\boldsymbol{x_k} - \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}})\right)\right\}, \tag{2.11}$$

where $\boldsymbol{\lambda_k}$ is a vector of Lagrange multipliers at time t_k, known as the discrete adjoint variables. It is the functional $\mathcal{L}$ that we now want to minimise, as a minimum of $\mathcal{L}$ will minimise J_o as well, whilst also imposing the necessary constraint (2.10). The minimum of $\mathcal{L}$ can be found by differentiating $\mathcal{L}$ with respect to each of the variables $\boldsymbol{x_k}$ and $\boldsymbol{\lambda_k}$ and setting the results to zero, i.e. we require the first variation of $\mathcal{L}$ to be zero.

Taking first the variation on $\boldsymbol{x_k}$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{x_k}} = -\boldsymbol{H_k}^T\boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})) + \boldsymbol{\lambda_k} - \boldsymbol{M_k^T}\boldsymbol{\lambda_{k+1}},$$

where $\boldsymbol{H_k} = \frac{\partial h_k}{\partial \boldsymbol{x_k}}\Big|_{\boldsymbol{x_k}}$ and $\boldsymbol{M_k} = \frac{\partial \boldsymbol{m_k}}{\partial \boldsymbol{x_k}}\Big|_{\boldsymbol{x_k}}$ (also known as the Jacobians), and then the variation on $\boldsymbol{\lambda_k}$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda_k}} = \boldsymbol{x_k} - \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}}),$$

leads to the Euler-Lagrange equations

$$-\boldsymbol{H_k}^T\boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})) + \boldsymbol{\lambda_k} - \boldsymbol{M_k^T}\boldsymbol{\lambda_{k+1}} = 0, \tag{2.12}$$

$$\boldsymbol{x_k} - \boldsymbol{m_{k-1}}(\boldsymbol{x_{k-1}}) = 0. \tag{2.13}$$

Equation (2.13) gives the constraint equation (2.10). Integrating equation (2.12) backwards in time from $t_{\mathcal{T}}$ to t_0 gives the adjoint equations

$$\boldsymbol{\lambda_{\mathcal{T}+1}} = 0 \tag{2.14}$$

$$\boldsymbol{\lambda_k} = \boldsymbol{M_k}^T \boldsymbol{\lambda_{k+1}} + \boldsymbol{H_k}^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})), \quad k = \mathcal{T}, \ldots, 0. \tag{2.15}$$

The gradient of J_o at the initial time is then given by

$$\nabla J_o(\boldsymbol{x_0}) = -\boldsymbol{\lambda_0}. \tag{2.16}$$

This can now be used in an iterative equation that requires the gradient of J to be calculated. $\boldsymbol{M}$ is known as the forward linear model and $\boldsymbol{M}^T$ as the adjoint model.

Equation (2.16), found using Lagrange multipliers, is the same as the result found using linear algebra. This is shown in Section 2.3.1.

This calculus of variations approach to formulating 4D-Var is illuminating in understanding the construction of 4D-Var as a constrained minimisation but is less helpful in understanding the practical implementation. We now go on to consider the implementation of 4D-Var using the linear algebra formulation.

2.3 Practical implementation of 4D-Var

In order to implement a 4D-Var system we must minimise the discrete cost function (2.2),

$$J(\boldsymbol{x_0}) = \frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{x^b})^T \boldsymbol{B}^{-1}(\boldsymbol{x_0} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k}))^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - h_k(\boldsymbol{x_k})),$$

to find our analysis. However, 4D-Var is a nonlinear constrained optimization problem which is very difficult to solve for the general case. Fortunately it can be greatly simplified using the tangent linear hypothesis. The tangent linear hypothesis allows the cost function to be linearised and solved using a method known as Incremental 4D-Var [16]. It is Incremental 4D-Var that is most often used operationally in NWP.

In this thesis however, we use a linear model (Chapter 7) and therefore the cost function does not need to be linearised and can simply be written as

$$\begin{aligned} J(\boldsymbol{x_0}) &= \frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{x^b})^T \boldsymbol{B}^{-1}(\boldsymbol{x_0} - \boldsymbol{x^b}) \\ &\quad + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - \boldsymbol{H_k x_k})^T \boldsymbol{R_k}^{-1}(\boldsymbol{y_k} - \boldsymbol{H_k x_k}), \\ &\overset{def}{=} J_b + J_o, \end{aligned} \tag{2.17}$$

and the constraint is

$$\boldsymbol{x_k} = \boldsymbol{M_{k-1} x_{k-1}}. \tag{2.18}$$

where J_b is defined to be the background term and J_o is defined to be the observation term.

2.3.1 Minimisation of the cost function J

To minimise the cost function we require a minimisation algorithm. Possible choices include the quasi-Newton or conjugate gradient methods [64]. Detailed descriptions of these methods can be found in [82], or [33, Sections 9.3, 10.2], for example, and are therefore not described here. However, these types of gradient-descent algorithms will require the value of J and its gradient ∇J to be calculated at each iteration and we therefore need to consider how these can be evaluated.

The evaluation of the cost function is relatively straight forward by simply calculating equation (2.17) as shown. J_b can be calculated directly. To calculate J_o we must simply use our tangent linear model $\boldsymbol{M_k}$ to evolve our solution forward in time. We can then calculate and store

$$\boldsymbol{d_k} = \boldsymbol{R_k}^{-1}\left(\boldsymbol{y_k} - \boldsymbol{H_k x_k}\right), \tag{2.19}$$

where $\boldsymbol{d_k}$ are known as the normalised departures. These departures can then be used to calculate J_o, as a series of contributions.

$$J_o(\boldsymbol{x}) = \frac{1}{2}\sum_{k=0}^{\mathcal{T}} J_{ok}(\boldsymbol{x}), \tag{2.20}$$

where

$$J_{ok}(\boldsymbol{x}) = \left(\boldsymbol{y_k} - \boldsymbol{H_k x_k}\right)^T \boldsymbol{d_k}. \tag{2.21}$$

The evaluation of the gradient can also be done in two parts. The gradient of J_b can simply be found by calculating ∇J_b directly,

$$\nabla J_b = \boldsymbol{B^{-1}}\left(\boldsymbol{x_0} - \boldsymbol{x_b}\right). \tag{2.22}$$

To calculate ∇J_o directly would be computationally unfeasible as it would require N forward model runs. The adjoint allows ∇J_o to be calculated with just one adjoint model run. We first need to factorise

$$\begin{aligned}\nabla J_o(\boldsymbol{x}) &= \frac{1}{2}\sum_{k=0}^{\mathcal{T}} \nabla J_{ok}(\boldsymbol{x}),\\ &= -\sum_{k=0}^{\mathcal{T}} \boldsymbol{M_1}^T \ldots \boldsymbol{M_k}^T \boldsymbol{H_k}^T \boldsymbol{d_k},\\ &= -\{\boldsymbol{H_0^T d_0} + \boldsymbol{M_1^T}[\boldsymbol{H_1^T d_1} + \boldsymbol{M_2^T}[\boldsymbol{H_2^T d_2} +\\ &\qquad \boldsymbol{M_3^T}[\boldsymbol{H_3^T d_3} + \cdots + \boldsymbol{M_{\mathcal{T}}^T H_{\mathcal{T}}^T d_{\mathcal{T}}}]\cdots]]\}. \end{aligned} \tag{2.23}$$

Equation (2.23) can now be calculated from right to left by initialising the adjoint variable $\boldsymbol{\lambda}$ to zero at the final time, $\boldsymbol{\lambda_{\mathcal{T}+1}} = 0$ (as in equation (2.14)). We then step backwards through the timesteps, at each step adding the forcing term $\boldsymbol{H_k^T d_k}$ to $\boldsymbol{\lambda}_k$, before applying the adjoint model to give $\boldsymbol{\lambda}_{k-1}$. Hence equation (2.23) becomes [12]

$$\nabla J_o(\boldsymbol{x}) = -\boldsymbol{\lambda}_0. \tag{2.24}$$

The gradient of J_o has been found here using linear algebra. The result (equation (2.24)) is the same as that found in Section 2.2 (equation (2.16)) derived by the method of Lagrange multipliers.

Here the adjoint model is defined as the transpose of the tangent linear model, i.e. $\boldsymbol{M}^T$. Operationally however, the adjoint is not constructed explicitly from the transpose of the tangent linear model matrix. The adjoint model, and how it is tested, is now considered in the next section.

2.3.2 The adjoint model

Operationally, the discrete model is linearised to give the tangent linear model and then the discrete adjoint equations are constructed from these discrete linearised equations. The adjoint is derived directly from the tangent linear model code using an 'automatic adjoint' method [29].

However, we can also use the matrix form of the discrete tangent linear equations to find the transpose. Due to the (small) size of the system to be used in this thesis, it is perfectly feasible here to simply transpose the matrix $\boldsymbol{M}$ to generate the adjoint, so that is what is done in our scheme (Chapter 5).

Once the adjoint is constructed, it needs to be tested. There are two different aspects which need to be tested and these are done using separate methods.

The Adjoint test

When the adjoint equations have been derived directly from the model code, the adjoint code needs to be tested to verify it is producing the correct adjoint. To do this we use the definition

$$\langle \boldsymbol{A}\boldsymbol{b}, \boldsymbol{c} \rangle = \left\langle \boldsymbol{b}, \boldsymbol{A}^T\boldsymbol{c} \right\rangle , \tag{2.25}$$

where $\boldsymbol{A}$ is a linear operator, $\boldsymbol{A}^T$ is its adjoint, $\boldsymbol{b}$ and $\boldsymbol{c}$ are vectors and the brackets $\langle \ldots, \ldots \rangle$ denote an inner product [29].

Therefore, for our model $\boldsymbol{M}$ and its adjoint $\boldsymbol{M}^T$ we can use equation (2.25) as a test by first applying the model to the initial conditions $\boldsymbol{\delta x_0}$ to produce a final state, then applying the adjoint to this final state.

If the adjoint is correct we should get

$$\langle \boldsymbol{M\delta x_0}, \boldsymbol{M\delta x_0} \rangle - \left\langle \boldsymbol{\delta x_0}, \boldsymbol{M}^T\boldsymbol{M\delta x_0} \right\rangle = 0.$$

The Gradient test

We must verify that the adjoint produces the correct gradient of the cost function. To do this we use the gradient test, a method already established to test adjoint models, for example [53], [63].

A Taylor expansion of the cost function J gives

$$J(\boldsymbol{x} + \alpha\overline{\boldsymbol{\delta x}}) = J(\boldsymbol{x}) + \alpha\overline{\boldsymbol{\delta x}}^T \nabla J(\boldsymbol{x}) + O(\alpha^2), \tag{2.26}$$

where α is a small scalar and $\overline{\boldsymbol{\delta x}}$ is a vector of unit length. This formula can be rearranged to give a function of α

$$\phi(\alpha) = \frac{J(\boldsymbol{x} + \alpha\overline{\boldsymbol{\delta x}}) - J(\boldsymbol{x})}{\alpha\overline{\boldsymbol{\delta x}}^T \nabla J(\boldsymbol{x})} = 1 + O(\alpha). \tag{2.27}$$

We should expect to get values of $\phi(\alpha)$ close to 1 for values of α that are small but not too close to machine zero. We should also get values close to zero for the residual $\phi(\alpha) - 1$. This can be displayed clearly by plotting a graph of $\log(\alpha)$ against $\log(\phi(\alpha) - 1)$.

2.3.3 Error covariance matrices

To run a 4D-Var scheme we need to define the error covariance matrices.

The matrix $\boldsymbol{R}$

The matrix $\boldsymbol{R}$ is the error covariance matrix associated with the observation errors. Operationally in NWP, the matrix $\boldsymbol{R}$ is assumed to be diagonal and all the cross-covariances are set to be zero, i.e. the errors are assumed to be uncorrelated. In reality however, this assumption often does not hold. For example, with errors coming from forward modelling error. However, the error correlations are difficult to estimate and the computational cost of computing the cost function and its gradient is much cheaper, in terms of matrix-vector multiplications, if $\boldsymbol{R}$ is diagonal [12].

The matrix $\boldsymbol{B}$

The matrix $\boldsymbol{B}$ is the error covariance matrix associated with the background errors. There are many techniques which have been developed to estimate the matrix $\boldsymbol{B}$. The favoured method has been the NMC (National Meteorological Centre) method which compares forecasts of different lengths, valid at the same time. A review of the matrix $\boldsymbol{B}$, including its role within Var assimilation, its mathematical properties and methods of estimating it, can be found in [6].

2.3.4 Control variable transforms

To minimise the cost function $J(\boldsymbol{x})$ (equation (2.2)) we vary the components of $\boldsymbol{x}$ iteratively. The vector $\boldsymbol{x}$ here is known as the control variable.

In Section 2.3.1 we demonstrated how to minimise the cost function with respect to $\boldsymbol{x}$. However, this method still leaves the problem of representing and inverting the matrix $\boldsymbol{B}$. For NWP this is simply infeasible, as the matrix $\boldsymbol{B}$ is typically of $\mathrm{O}(10^7 \times 10^7)$. To overcome this problem, the cost function is instead posed in terms of a different control variable, by means of a control variable transform [7]. This new control variable is designed to have beneficial properties. The principle is for it to simplify the background term of the cost function.

Operationally, the control variable transform is designed to convert from model variables to new variables which have no correlations between parameters and are also uncorrelated in both the vertical and the horizontal.

A discussion on control variable transforms, including those used operationally by the UK Met Office and ECMWF (European Centre for Medium range Weather Forecasting) can be found in [7] or [56].

2.4 Summary

We have described several types of data assimilation, sequential and variational, 3-dimensional and 4-dimensional. We have introduced examples of these different types of scheme and discussed differences in their implementation. We have also considered the benefits of the different schemes. 3D schemes have the advantage that they are generally computationally cheaper to run but the 3D assumption can cause information to be lost from the observations. 4D schemes allow observations to be treated more realistically as they can be distributed in time and assimilated at the time they are valid. However, this addition of a time dimension does cause 4D methods to be computationally more expensive.

Having considered the different schemes available, we went on to discuss 4D-Var in more detail. We considered its mathematical formulation as well as its practical implementation. This was done because 4D-Var is the scheme that will be used throughout the rest of this thesis.

We now go on to introduce the discrete Fourier transform (DFT) and its power spectrum in Chapter 3. We demonstrate features of the power spectrum and how they enable us to use the DFT as a tool to understand the analysis generated by the data assimilation.

Chapter 3

The discrete Fourier transform

In this chapter we introduce the discrete Fourier transform (DFT) and the power spectrum. The DFT can be an extremely useful method for understanding the analysis generated by the data assimilation. We demonstrate features of the power spectrum and how they relate to properties of the DFT, and thus how they enable us to use the DFT as a tool to understand the analysis.

3.1 Definition

Sines, cosines and imaginary exponentials have the property that they are orthogonal over a series of discrete, equally spaced points [37, p.25],

Lemma 3.1

$$\frac{1}{N}\sum_{j=0}^{N-1} e^{-is\xi_j} = \begin{cases} 1 & \textit{if } s = Nm,\ m \in \mathbb{Z} \\ 0 & \textit{otherwise,} \end{cases}$$

where $\xi_j = 2\pi j/N$

Using this orthogonality we can construct the discrete Fourier transform (DFT) for a function f defined at discrete values ξ_j over the domain (a, b) where $j = 0, ..., N$ and $\xi_0 = a$, $\xi_N = b$. This is discussed by [1, p. 788-9], amongst others.

The DFT of a function f_j (where $f_j = f(\xi_j)$) is defined to be

$$DFT\,(f_j) = \widehat{f}_k = \sum_{j=0}^{N-1} f_j e^{-i2\pi jk/N}, \qquad k = 0, 1, \ldots, N-1, \tag{3.1}$$

where k is the wavenumber and N is the number of gridpoints [13, p.52]. The inverse transform (IDFT) is

$$f_j = \frac{1}{N}\sum_{k=0}^{N-1} \widehat{f}_k e^{i2\pi jk/N}, \qquad j = 0, 1, \ldots, N-1. \tag{3.2}$$

This discrete Fourier transform (DFT) can be calculated on a computer using a method called the Fast Fourier Transform (FFT). This FFT reduces the number of computer operations needed from $O(N^2)$ to $O(N\log(N))$ [1, p.791]. For small values of N this is not so significant but for large values of N this reduction in computational expense is essential in making the calculations possible.

3.2 A simple example

In order to understand better the DFT we begin by studying it for a simple example, namely the sine wave $f_j = \sin(\kappa\xi_j)$.

3.2.1 Calculating the DFT of $\sin(\kappa\xi_j)$ analytically

To investigate the structure of the DFT we first calculate it analytically for the sine wave $f_j = \sin(\kappa\xi_j)$.

Letting $f_j = \sin(\kappa\xi_j)$, $0 \leq \kappa \leq N-1$ where $\kappa \in \mathbb{N}$ is the wavenumber, equation (3.1) becomes

$$\widehat{f}_k = \sum_{j=0}^{N-1} \sin(\kappa\xi_j) e^{-i\xi_j k}. \tag{3.3}$$

Writing $\sin(\kappa\xi_j) = \frac{1}{2i}\left(e^{i\kappa\xi_j} - e^{-i\kappa\xi_j}\right)$ this becomes

$$\widehat{f}_k = \frac{1}{2i}\sum_{j=0}^{N-1}\left(e^{i\kappa\xi_j} - e^{-i\kappa\xi_j}\right) e^{-i\xi_j k},$$

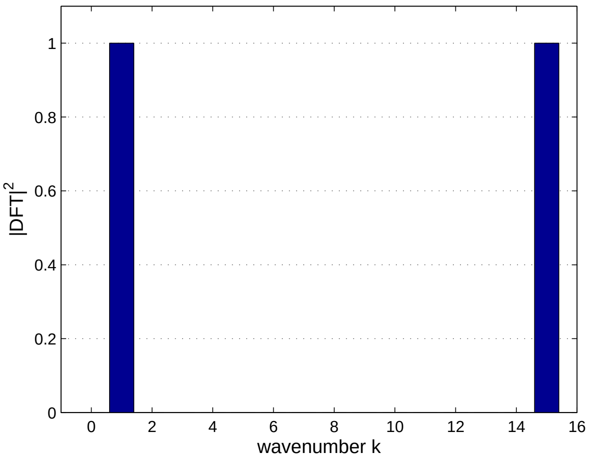


Figure 3.1: The power spectrum of $\sin(\xi_j)$, scaled by the factor $2/N$.

which can be rearranged to give

$$\widehat{f_k} = \frac{1}{2i}\sum_{j=0}^{N-1} e^{-i\xi_j(k-\kappa)} - \frac{1}{2i}\sum_{j=0}^{N-1} e^{-i\xi_j(k+\kappa)}. \tag{3.4}$$

In order to evaluate equation (3.4) we use the fact that the complex exponential function obeys the orthogonality relation given by Lemma (3.1). Using Lemma (3.1) with $s = k - \kappa$ and then $s = k + \kappa$ and noting that $\frac{1}{i} = -i$, equation (3.4) becomes

$$\widehat{f_k} = \begin{cases} \frac{-iN}{2} & \text{if } k = \kappa, \\ \frac{iN}{2} & \text{if } k = N - \kappa, \\ 0 & \text{otherwise.} \end{cases} \tag{3.5}$$

3.2.2 The power spectrum

In order to demonstrate the propeties of the DFT which are of interest here, we first need a convenient method of analysing the DFT coefficients. To do this we consider the square of their modulus. A suitably scaled plot of these moduli is called a power spectrum [25, p.53]. We apply a scaling factor of $2/N$ to the DFT coefficients, where N is the number of gridpoints. This results in a standard trigonometric wave (i.e. one with an amplitude of one) having an amplitude of one in the power spectrum. For example, Figure 3.1 shows the power spectrum of $\sin(\xi_j)$ on the domain $x = [0, 1)$, where $\xi_j = 2\pi x_j$, $j = 0, \cdots, N-1$, $x_j = j/N$ and $N = 16$.

By setting $\kappa = 1$ in equation (3.5), we see that it agrees with the power spectrum of $\sin(\xi_j)$ plotted in Figure 3.1, which has amplitude at $k = 1$ and $k = N - 1$ and is zero everywhere else. By applying the scaling factor $2/N$ to equation (3.5) we also obtain an amplitude of one.

From this simple example we can begin to see the usefulness of the power spectrum. The power spectrum provides a method to observe the features of the DFT coefficients in an illustrative way.

There are many known properties of the DFT and descriptions of these can be found in [13, Section 4.1 and 4.2], [25, p.49] and [72, p.53-56] amongst others. We will explore a few of these properties and relate them to the features we see in the power spectrum. We will demonstrate how the DFT can be used as a tool to understand the analysis generated by the data assimilation.

3.3 Properties of the DFT

1. **Linearity**

   The DFT is a linear function [25, p.49], such that, for a function

   $$y_j = af_j + bg_j,$$

   where a and b are scalars, the DFT of y_j is

   $$\widehat{y_k} = a\widehat{f_k} + b\widehat{g_k}.$$

2. **Symmetry of the Complex Conjugate**

   For a real sequence f_j with $j = 0, \cdots, N-1$, its DFT will, in general, be a sequence of N complex numbers. In particular $\widehat{f_k}$ and $\widehat{f}_{N-k}$ are related by

   $$\widehat{f}_{N-k} = \widehat{f}_k^*$$

   for $k = 0, \cdots, N-1$, where $*$ denotes the complex conjugate [13, p.56].

3. **Phase**

   In general, the DFT of a function f_j can be written as

   $$\widehat{f_k} = \widehat{f_k^R} + i\widehat{f_k^I},$$

   where $\widehat{f_k^R}$ is the real part and $\widehat{f_k^I}$ is the imaginary part.
   However, for an even function g_j, its DFT can simply be written as

   $$\widehat{g_k} = \widehat{g_k^R}.$$

   For an odd function h_j, its DFT can simply be written as

   $$\widehat{h_k} = i\widehat{h_k^I}.$$

   i.e. even functions transform to real parts only, and odd functions transform to imaginary parts only [72, p.53]. This can easily be shown from the definition (3.1).

3.4 Features of the power spectrum

3.4.1 Location of amplitude

If we consider the sine and cosine waves as a basis for function space, then a non-zero value of the DFT coefficient is determined by the wavenumbers which contribute components to the function being transformed. A non-zero magnitude in the power spectrum at wavenumber κ means the function being transformed contains a component of the wave with wavenumber κ.

In particular, for functions that are simply a linear superposition of sine waves, this means the power spectrum will pick out the wavenumbers of those waves in the function. For a single sine wave example, this property is clearly demonstrated in the calculation shown in Section 3.2.1 and in Figure 3.1. As we see, the transform of $\sin(\xi_j)$ has non-zero magnitude at $\kappa = 1$ and at $\kappa = 15 = N - 1$. This second location is caused by the second property given in Section 3.3, the symmetry property of the complex conjugate. It causes a function to have non-zero transform values in two places, not one. One location corresponds to the wavenumber κ and one corresponds to the wavenumber $N - \kappa$, as seen in Figure 3.1.

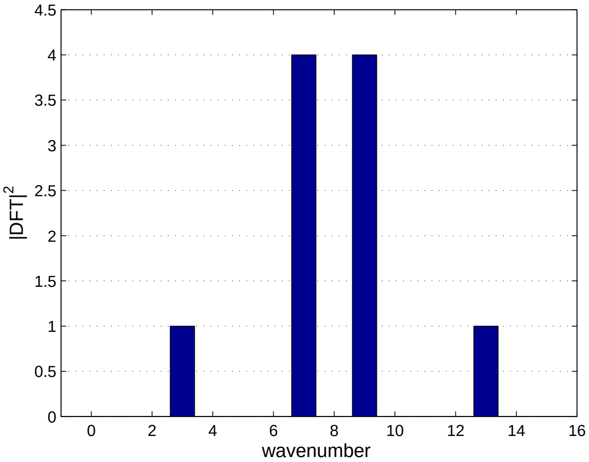


Figure 3.2: The power spectrum of $\sin(3\xi_j) + 2\sin(7\xi_j)$, scaled by the factor $2/N$.

Figure 3.2 shows the power spectrum of the wave $y = \sin(3\xi) + 2\sin(7\xi)$, i.e. a linear combination of sine waves. As we see, we can again pick out the wavenumbers present in the function by the non-zero magnitude in the power spectrum. This is the case for a function with more than one wavenumber component due to the linearity property of the DFT given in Section 3.3.

This feature of the power spectrum demonstrates one way we can use the DFT to understand the analysis generated by the data assimilation. The power spectrum allows us to identify the wavenumbers that form the components of the analysis.

3.4.2 Magnitude

If we again consider the sine and cosine waves as a basis for function space, then from the magnitude in the power spectrum we can identify scaling factors on the components of a function.

From the linearity property of the DFT given in Section 3.3, if a function has a scaling factor then the DFT coefficients also have the same scaling factor (which will be squared in the power spectrum). Therefore a magnitude not equal to one at wavenumber κ in the power spectrum signifies a scaling factor on the function component with wavenumber κ. This can be seen in Figure 3.2 where the wave $y = \sin(3\xi) + 2\sin(7\xi)$ has a scaling factor of 2 on the wave with wavenumber $\kappa = 7$ and this corresponds to an amplitude of 4 in the power spectrum.

This property of the power spectrum allows us to use the DFT to identify scaling factors in an analysis produced by the data assimilation.

3.4.3 Phase

As stated in Section 3.3, odd functions transform to imaginary parts only and even functions transform to real parts only. Functions can be odd or even, or can be made up of a combination of both odd and even parts. By considering the real and imaginary parts of the DFT separately we can understand better the phase of a function. For example, $y_j = \sin(\kappa\xi_j)$ is an odd function and therefore has purely imaginary DFT coefficents. This is demonstrated in the calculation done in Section 3.2.1. However, if we now consider a sine wave with a phase shift we can see another useful property of the DFT.

To illustrate how the DFT can be used to understand phase we take the function $y_j = \sin(\xi_j + \phi)$, where ϕ represents a phase shift. We take $\phi = \frac{\pi}{4}$. Using trigonometric angle summation formulae we can rewrite our function as

$$\begin{aligned} y_j &= \sin(\xi_j + \frac{\pi}{4}), \\ &= \frac{1}{\sqrt{2}}\left(\sin(\xi_j) + \cos(\xi_j)\right). \end{aligned}$$

As we see, our function is a sum of an odd and an even function.

We now consider the real and imaginary parts of the DFT of our function. Figure 3.3 is a plot of the square of the absolute value of the real (red) and imaginary (green) parts of the DFT. As can be seen half the amplitude is in the real part and half is in the imaginary part. This corresponds to our function being a sum of an odd and an even part. It is an equal split between real and imaginary because the function is an equal split between odd and even parts. The DFT can identify differences in phase and this is easily seen in the power spectrum.

The capability to use the the power spectrum to identify differences in phase can be used as a method of understanding the analysis generated by the data assimilation.

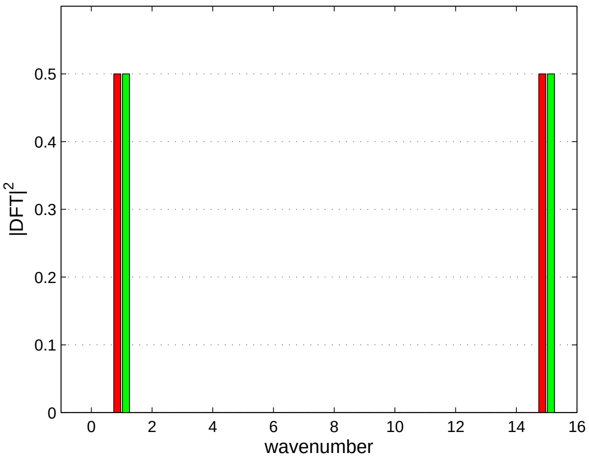


Figure 3.3: The power spectrum of the real (red) and imaginary (green) parts of the DFT of $\sin(\xi_j + \pi/4)$, scaled by the factor $2/N$.

3.5 Summary

We have shown that it is possible to learn a lot of information about a solution by simply performing a DFT and considering the features of its power spectrum in relation to the known properties of the DFT. By considering the sine and cosine waves as a basis for function space, we have shown that the wavenumber and pattern, amplitude and phase of the components of a function can all be found from the DFT and its power spectrum. We have also demonstrated methods for how this can be done. These methods have been demonstrated here on basic sine wave examples but can easily be applied to more complicated functions.

Chapter 4

Limited Area Models

There are many different limited area models (LAMs) currently being run operationally at weather centres around the world. We begin by introducing some of these operational schemes. We go on to discuss problems associated with the limited area nature of high resolution models such as the need for boundary conditions and the problem of having atmospheric phenomena with length scales longer than the domain of the model.

4.1 Operational models

In order to produce more accurate forecasts of high impact weather, weather centres around the world are developing and running high resolution, limited area forecast models. The high resolution allows a more accurate representation of the dominant motions at the convective scale and therefore a more accurate convective scale forecast. Data assimilation schemes are also being developed for these high resolution, limited area models.

4.1.1 The LAM models

Many different weather centres run limited area, mesoscale models operationally. These models vary in domain size, resolution and model specifications.

- The Unified model (UM) at the UK Met Office is run on limited area domains as well as globally [79].

- The WRF (Weather Research and Forecasting) model is a multiagency collaboration [78]. Current major WFR partners include NCAR, the National Oceanic and Atmospheric Administration's (NOAA)'s National Center for Environmental Prediction (NCEP), NOAA Forecast Systems Laboratory (FSL), AFWA and the Center for the Analysis and Prediction of Storms (CAPS) at Oklahoma University. The WRF model is a convective-mesoscale model for use by both research and industry.

- The ALADIN/France model is a LAM run operationally at Météo-France [26]. Also being developed is the AROME (The Application of Research to Operations at MesoscalE) system, the next generation LAM NWP system for Météo-France [26].

- HIRLAM (High Resolution Limited Area Model), a collaboration between the National Meteorological services in Denmark, Finland, Iceland, Ireland, Netherlands, Norway, Spain and Sweden, runs HIRLAM-5 [38], a high resolution NWP system for the synoptic scale. It is also developing HARMONIE for the mesoscale, in collaboration with ALADIN.

- The Japan Meteorological Agency (JMA) has operated a mesoscale numerical prediction (Meso NWP) since March 2000 in aid of disaster relief [41]. Initially the mesoscale model (MSM) in the meso NWP system was a hydrostatic model but was upgraded to the JMA non-hydrostatic model (JMA-NHM) in September 2004.

These are just some of the models run around the world, both operationally and for research purposes. All of the models have their individualities but also share some common themes.

One theme shared by mesoscale models is the increasing resolution they are being run at. At the UK Met Office the NAE (North Atlantic and Europe model) is run at 12km resolution and a 4km resolution domain is also operational [51]. This 4km version is currently being upgraded to 1.5km. WRF has a 1-10km resolution range [8].

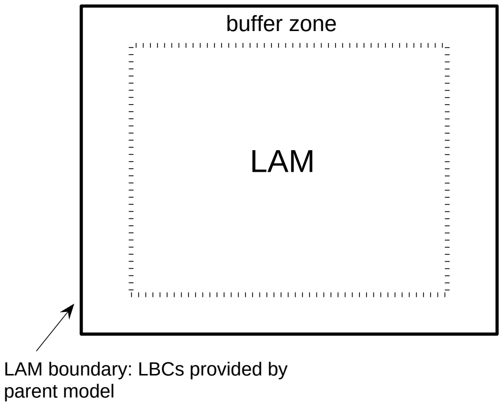


Figure 4.1: Schematic of the LAM domain and buffer zone.

HIRLAM-5 is run at 5-15km resolution [39] and HARMONIE has a target resolution of 2.5km [40]. Aladin has a horizontal resolution of about 9.5km and AROME will run at 2.5km gridmesh [26]. The MSM at JMA currently has a 5km resolution [46].

Another shared property of all the models is that their lateral boundary conditions (LBCs) are provided from a larger, courser resolution (parent) model. In order to relax the solution on the interior of the LAM domain to the values prescribed by the parent model at the boundaries, a buffer zone is implemented at the boundaries of the LAM, as illustrated in Figure 4.1. This relaxation is often done via Davies relaxation[1] [20], [21].

Operationally, the Davies Relaxation is implemented in a discrete form and the solution at a gridpoint is given by

$$u_i^{buffer} = (1 - \alpha_i)u_i^L + \alpha_i u_i^P, \tag{4.1}$$

where u is the model variable, $i = 0, \ldots, b-1$ are the LAM gridpoints within the buffer zone with $i = 0$ corresponding to the external boundary, u_i^{buffer} is the value of u in the buffer zone, u_i^L is the value of u calculated by the limited area model, u_i^P is the value of u coming from the parent model and α_i is the relaxation term at gridpoint i. On the external boundary $\alpha_1 = 1$, and $\alpha_i = 0$ for all lines beyond the buffer zone.

[1] MSM at JMA utilises a Rayleigh damping term [46] instead but this damping term is closely related to Davies relaxation.

Although all the models utilise this relaxation across a buffer zone, the structure of the relaxation term α_i and the width of the buffer b varies between models. For example, in the WRF system $b = 4$ and α is a simple linear interpolation [78] whereas in HIRLAM-5, $b = 10$ is the default in the reference system and the default value of α is $\alpha_i = \frac{1}{2}\left\{1 + \cos[\pi i/b]\right\}$ [38]. As well as a relaxation zone, HIRLAM also implements an extension zone to obtain periodicity on the LAM domain [35]; this will be discussed in Section 4.2.1.

All the LAMs get their LBCs from a parent model. However, the frequency with which they are provided differs between the models. In the WRF system the LBCs are specified by the parent model at every coarse-grid time step [78]. In contrast, in MSM the LBCs are only given at the initial time and at the end of the assimilation window and are interpolated between these two times [41]. This interpolation could degrade the accuracy of the model as it forces all variations at the boundaries to be equal in size and monotonic, when in reality the values vary nonuniformly and may oscillate. This is illustrated in Figure 4.2. Points A and B are the boundary value u_0 at the initial time and end of the assimilation window respectively. The dotted line is the linear interpolation between the points and the red dots are the boundary values obtained from this interpolation. The solid blue line shows the model value at the boundary over the assimilation window and the blue dots show the boundary conditions obtained from this trajectory. As can be seen, these two sets of LBC values can be significantly different, which would have an effect on the LAM using them. Errors in the LBCs may also propagate inwards over the assimilation window so that even though only the boundaries of the LAM are forced to agree with the parent model, the solution on the interior of the LAM has been degraded. By providing LBCs at every coarse-grid timestep some of the natural nonuniformity of the variation is maintained and may help provide a more accurate analysis across the LAM domain.

All the models discussed so far share one more common theme, they run on a grid with a constant resolution. However, as well as a nested-grid approach, there has been interest in developing stretched/variable resolution grids for high resolution NWP [3]. A global stretched/variable resolution grid with enhanced resolution over the region of interest has been used for regional scale forecasts and climate modelling at the Canadian Meteorological Centre (CMC) [15], [27]. A stretched/variable resolution grid has the

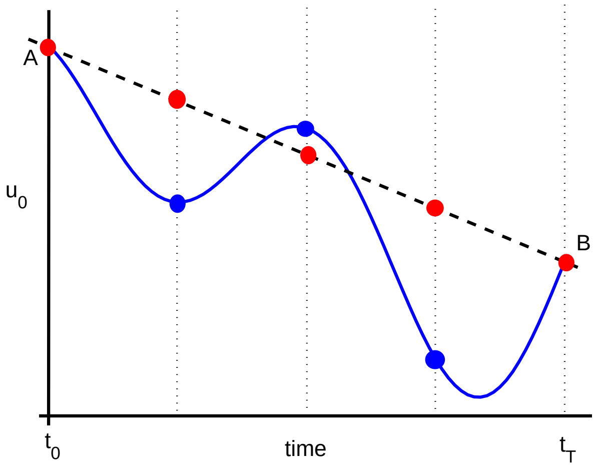


Figure 4.2: Comparing methods of providing boundary values for the LAM. Points A and B are the boundary value u_0 at the initial time and end of the assimilation window respectively. The dotted line is the linear interpolation between the points and the red dots are the boundary values obtained from this interpolation. The solid blue line shows the model value at the boundary over the assimilation window and the blue dots show the boundary conditions obtained from this trajectory.

benefit that it does not require LBCs and provides self-consistent interactions between the global and regional scales of motion. However, the variable resolution has its own complications. In particular, there is the question of how to perform data assimilation on a variable resolution grid. At CMC the 3D-Var analysis increments are generated on a fixed resolution grid and then interpolated to the variable resolution.

4.1.2 The data assimilation schemes

Variational data assimilation (Var) is the method of choice for global models in many numerical weather prediction centres and is expected to remain so for the foreseeable future [7]. For LAMs, Var is also the method of choice. HIRLAM [43] and the JMA model run 4D-Var with the option to run as 3D-Var. WRF currently runs with 3D-Var, while 4D-Var is under developement [78]. Aladin at Météo-France and both the NAE and 4km UK model at the Met Office have 3D-Var systems.

Var is being and will be used with increasingly higher resolution models. There are already plans published by Météo-France for a 3D-Var-based high-resolution limited-area system with a model of 2.5km grid length [26], the Met Office 4km UK model is being upgraded to 1.5km and HARMONIE at HIRLAM, which has a target resolution of 2.5km, has a 3D-Var scheme with 4D-Var under development [40].

Although the methods of implementation may vary between the centres, most of the Var systems in use operationally on LAMs solve the same basic cost function. The Var used by JMA however is a notable exception. It contains an extra term J_c in the cost function [46]. This J_c is a penalty term to suppress spurious inertia-gravity waves [41]. At Météo-France there are also plans to introduce an additional term, J_k, to the standard cost function (This will be discussed in Section 4.2.1). This term would add the large-scale structures from a parent model as an extra source of information, by means of a weak constraint.

Having discussed some of the operational schemes, their similarities and differences and how they overcome some of the issues caused by the limited area domain size, we now go on to consider some of the problems associated with limited area models in more detail.

4.2 Problems caused by having a limited area domain

Only having a limited area can create some problems for both the data assimilation and the model itself.

4.2.1 Lateral boundary conditions

A significant problem caused by the limited area nature of storm-scale forecasting models is the need for lateral boundary conditions (LBC). A global model grid stretches over the whole globe so has no 'edges' and is simply periodic. For a limited area model (LAM) this is not the case. The grid now has 'edges' so as well as initial conditions we also need to have LBCs for both the model and the data assimilation. These LBCs can have several associated problems.

Lateral boundary conditions for the model

One property the research models at the world's weather centres all have in common is that they take these LBCs from another model which covers a larger domain. This model is refered to by various names but here will be called the 'parent' model. The

LBCs supplied by the parent model can be a major source of error in the LAM [65],[84]. This is due to several reasons.

1. The horizontal, vertical and temporal resolution of the parent model is generally coarser than that of the LAM and thus the LBCs have to be interpolated to the LAM grid at every timestep. Even if the parent model run providing the LBCs is perfect, there will be interpolation errors introduced when it is interpolated to the LAM resolution.

2. There may also be differences in model setup and parameterisation, even if the resolution is the same in both models. This will cause differences in the model outputs and these differences at the boundaries may cause spurious waves and gradients, which can influence the LAM interior.

3. The parent model may simply be wrong for some reason. It may have an atmospheric feature occuring in the wrong spatial location or developing too early or too late for example, and this error will be transmitted to the LAM at the boundary interface.

These problems and others are dicussed by [84]. Although these are problems associated with the LAM itself, they can also be exacerbated in the data assimilation. This is discussed in the next section.

One way to negate these problems is to simply locate the boundaries sufficiently far from the area of meteorological interest such that errors cannot be transmitted inwards from the boundary in the time window of the forecast [84]. For high resolution forecasting however, this is simply impractical as the domain can only be a limited size and there are insufficient computer resources to allow for boundaries to be sufficiently far away.

As discussed in Section 4.1.1, one method that is implemented operationally is the use of a ‘buffer zone’. The buffer zone blends the two solutions near the boundaries and therefore avoids any sharp jumps, which could arise because the models evolve independently, by smoothing any discrepencies between the models at the boundaries.

Lateral boundary conditions for the data assimilation

As well as being required for the model, LBCs are also required for the data assimilation algorithm. These LBCs share or exacerbate some of the problems already discussed but also cause others specific to the data assimilation.

The operational assimilation-forecast cycle has the following steps:

Step 1 Run a coarse resolution model to provide a forecast over a large domain.

Step 2 Run the LAM, from a previous LAM analysis, to provide a background forecast over the assimilation time window, using LBCs from the forecast in Step 1.

Step 3 Run the data assimilation to combine the LAM background forecast with observations to produce the LAM analysis.

Step 4 Run the LAM over the full forecast period, using the analysis from Step 3 and the LBCs from Step 1.

One problem is that the LAM analysis is unable to use observational information from outside its domain. Near the boundary this may be detrimental locally [2]. This is illustrated in Figure 4.3. Although the observation lies outside the area covered by the LAM domain, the $\boldsymbol{B}$ matrix spreads out the influence of this observation and would allow it to provide information at gridpoints inside the LAM domain. However, because the LAM cannot see this observation, the knowledge it infers about gridpoints within the LAM domain is also lost. It may also impact negatively if there is a large scale feature partially lying within the LAM domain [34]. In order to capture the feature properly information could be needed from outside the domain but this information is unavailable to the LAM.

Another problem is that the LBCs interpolated from the parent model have the potential to degrade the quality of the analysis produced by the LAM. The background error covariance matrix spreads information so errors at the boundaries will be spread inwards. During a 4D-Var assimilation window, the model dynamics will also spread the errors from the boundaries into the interior of the LAM, contaminating the analysis [65]. Furthermore, the analysis is updated separately from the LBCs, as illustrated in Steps

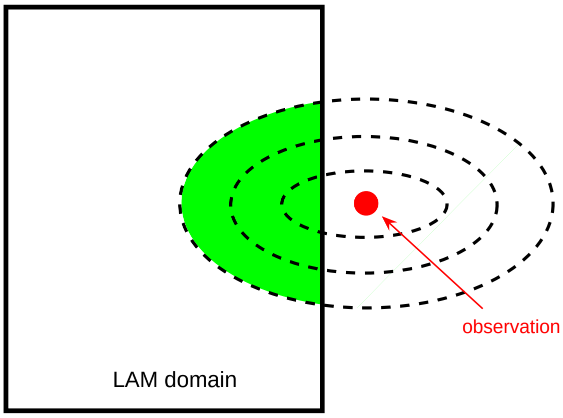


Figure 4.3: Diagram illustrating the observational information lost from observations outside the LAM domain. The red dot is an observation and the dotted lines indicate the spreading of this observational information by the matrix $\boldsymbol{B}$. The green area is the part of the LAM domain that would be influenced by the observation if it were visable to the LAM.

1-4 above. This can create inconsistencies between features represented in the LAM and the coarser model providing the LBCs. For example, the position of a weather front may be different in the two domains.

As discussed in Section 2.3.4, a control variable transform is used to remove horizontal correlations in the state variable. In global variational assimilation schemes this control variable transform results in a state variable defined in spectral space. However, applying a spectral transform on the LAM poses a problem as the solution on the LAM domain is not periodic. This leaves us with two choices for the treatment of the boundaries in the assimilation. One option is to analyse the boundaries and update the LBCs in the forecast. This option is implemented by HIRLAM [35]. To do it they utilise an extension zone, as shown in Figure 4.4. This extension zone is a region added to the edge of the LAM domain over which the solution on the LAM is made periodic. It makes it possible to have non-zero boundary increments on the LAM and have the periodic solution necessary to apply standard fast Fourier transforms for the transform of the control variable. However, a problem caused by updating the LBCs is that the assimilation time window is much shorter than the full forecast period. At the end of the assimilation window we have to revert to the LBCs from the coarse background forecast (from Step 1), potentially creating discontinuities in the LBCs.

The second option is to not analyses the boundaries and force them to remain the

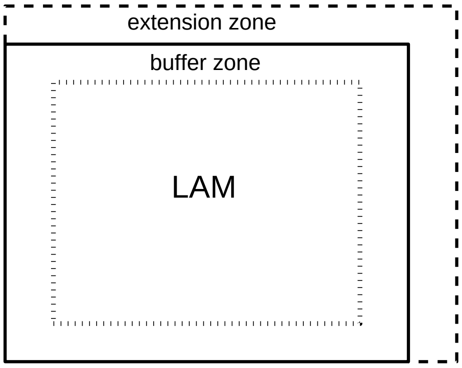


Figure 4.4: The extension zone of the HIRLAM model, as shown in [35]

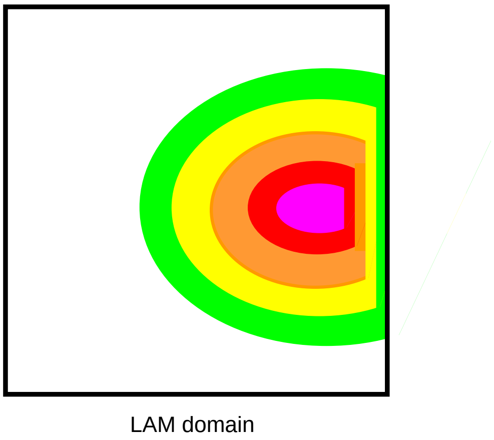


Figure 4.5: Diagram illustrating a feature being ‘cut-off’ by the LAM boundary. The colours indicate the size of the increment, white being zero increment and pink being the largest increment.

same as those of the parent. If we think in terms of increments, this second option can be described as having zero boundary conditions, as the boundaries are not allowed to change. A consequence of zero boundary conditions is that we have the periodic solution we need in incremental space to perform a spectral control variable transform. In particular, having zero boundary conditions mean that a sine transform can be used, as is implemented at the UK Met Office [55]. However, zero boundary conditions can result in phenomena being ‘cut-off’ or contained by the boundary when they should pass through it [2]. This is demonstrated in Figure 4.5.

One attempt by [2] to improve this problem of phenomena being ‘cut-off’ by the boundaries has been to modify the LAM background by adding on the analysis increments of the coarse analysis, filtered to scales greater than a specified length scale N. The analysis in the LAM is then only done for scales less than N km. Results presented in [2] show this can improve the increments in the region of the boundary but the increments at the boundary are still zero so phenomena are still being cut off. This method will not solve the problem when the LAM assimilation is run more frequently than the model providing the LBCs [2], as the results are produced assuming that the coarse model analysis is available at the time of the LAM analysis. Operationally, this may not be the case. However, it might be possible to use a coarse model forecast to provide the large scales, but this has not been tested.

An alternative method proposed by [34] attempts to improve the consistency of the LAM analysis with the LBCs provided by the parent model by including an extra term in the cost function. This extra term measures the distance between the lateral boundary values of the analysis and the LBCs from the global model in a low resolution geometry. This fits with the recommendations in [65] and [87] that the LBCs be included as part of the control variable in addition to the initial conditions. Slightly positive or neutral results were seen for the proposed method [34]. However, testing the method on a very high resolution, small domain size is still to be done and the method is only considered for a 3D-Var assimilation scheme; no comment is made as to how this method could be extended to a 4D-Var scheme. As with the previous method, this method is not currently operational.

Another problem caused by the limited domain size is that there may be features present in the atmosphere with lengthscales that are longer than the domain of the model. However, before we discuss this we first need to consider aliasing.

4.2.2 Aliasing

Aliasing is a phenomenon whereby small-scale waves are misinterpreted as much larger-scale waves. Following the description in [18, Appendix H], the problem is demonstrated in Figure 4.6. The red line is a plot of the wave $\sin(2\pi 9x)$ and the 9 black dots are observations taken at equally spaced intervals. As can be seen by the blue line, the 9 observations also fit the wave $\sin(2\pi x)$. With just these 9 points, there is no way to

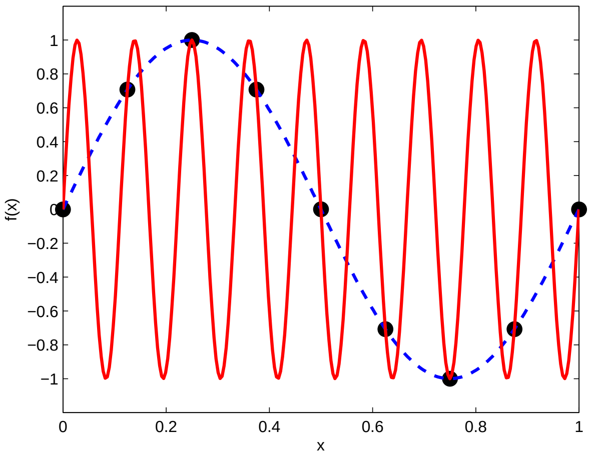


Figure 4.6: Diagram illustrating the problem of aliasing. The red solid line is $f(x) = \sin(2\pi 9x)$ and the blue dashed line is $f(x) = \sin(2\pi x)$. Black dots are observation points.

distinguish between the small-scale and the larger-scale wave. With eight grid lengths, as there are in Figure 4.6, only the first eight degrees of freedom can be extracted and any waves with higher wavenumber are misrepresented.

Aliasing can be explained algebraically if we consider the Fourier series. For a function $f(x_j)$ where $x_j = 2\pi j/N$ and $j = 0, ..., N-1$, the discrete Fourier modes are based on the points x_j, for which the $(n+Nm)$th mode (where m is an integer) is indistinguishable from the nth mode [37, p.31],

$$e^{i(n+Nm)x_j} = e^{inx_j} e^{i2\pi mj} = e^{inx_j}.$$

High-order modes will therefore be misrepresented as lower-order modes.

In an NWP model, this can mean that atmospheric phenomena at scales below the resolution of the model are aliased onto lower wavenumbers, causing errors at the longer scales. The phenomenon of aliasing with reference to data assimilation is discussed by [18, Appendix H]. The higher resolution of LAMs should reduce this aliasing effect compared with the parent model, as more small scales can be resolved. However, the LAM can also suffer a different aliasing problem caused by its limited area domain. This is now discussed in Section 4.2.3.

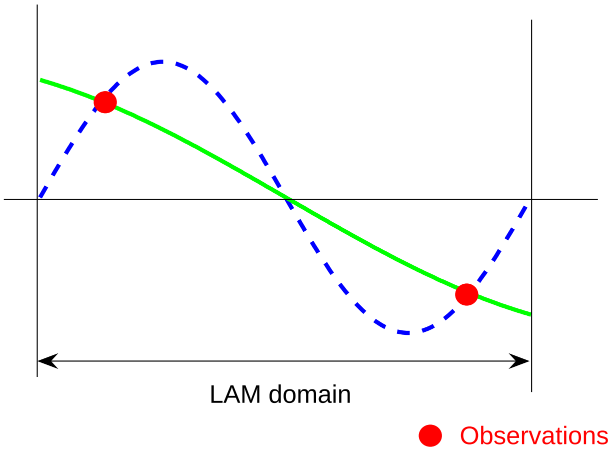


Figure 4.7: The long-wave problem. The long wave (green line) is longer than the domain of the LAM. Observations of this wave (shown as red dots) could be used by the data assimilation to correctly reconstruct the wave section, or they could be used to incorrectly reconstruct a wave with a shorter wavelength which fits in the domain of the LAM (blue dashed line).

4.2.3 The 'long-wave' problem

Another problem for a LAM which is a consequence of the limited area domain is that there may be atmospheric phenomena with lengthscales that are longer than the domain of the model. For example, a Rossby wave can have a length scale of several thousand kilometers compared to the 1152km domain of the Met Office 4km model.

The problem of having lengthscales longer than the domain of the model is referred to here as the 'long-wave' problem and these long lengthscales will be referred to collectively as 'long-waves'. Accurately representing these long-waves is important because meteorological processes are known to be multiscale phenomena and there are strong feedbacks between synoptic and convective scale behaviours [19]. It has been shown in case studies [74] that if the large scales are poorly represented then we cannot hope to generate an accurate forecast of the convective scales, even if we have captured the local dynamics well.

However it may be difficult to capture the long-waves correctly on the LAM domain as their wavenumbers are not among the range of the limited area model. Data assimilation may help as atmospheric measurements can contain a large range of scales. However, observations of the long-waves may be aliased onto shorter wavelengths. This is demonstrated in Figure 4.7 where only a section of the long-wave (shown in green) is contained within the LAM domain. If we have observations (shown as red dots) of this long-wave section, the data assimilation may correctly reconstruct the long-wave. It might however, interpret the observations as coming from a wave whose wavelength does fit within the LAM domain and reconstruct that wave instead (as shown in blue).

In contrast to the tradional view of aliasing described in Section 4.2.2, the long-wave problem can be considered as a ‘reverse’ aliasing problem, with long-wave information being aliased onto shorter wavelengths that are part of the spectrum of the domain.

4.3 Summary

In this chapter we discussed some of the operational limited area models currently in use around the world. We commented on the similarities and differences between the models and their data assimilation schemes as well as how they deal with some of the issues caused by having only a limited domain size.

We then went on to consider in more detail some of the problems caused by the limited domain size. We discussed problems caused by the need for boundary conditions and we also introduced the long-wave problem: the problem of having atmospheric phenomena with length scales longer than the domain of the limited area model.

It is important that a LAM analysis be able to capture both small and large scales, including scales longer than the domain of the model. In this thesis we aim to understand where information about the different scales is projected by the data assimialtion and to investigate a possible method for using this knowledge to improve the representation of large scales, including long-waves, in a 4D-Var LAM analysis whilst still accurately capturing smaller scale detail.

In Chapter 5 we go on to discuss the adaptations to the 4D-Var scheme introduced in Chapter 2, to enable it to be used for a limited area domain.

Chapter 5

A 4D-Var algorithm for a LAM domain

In this Chapter we introduce a general limited area model (LAM) domain and discuss the modifications needed to a standard 4D-Var scheme in order to run it on this limited area domain. We also discuss minimising the 4D-Var scheme in both physical and spectral 4D-Var space.

5.1 A parent domain and a nested LAM domain

We have two 1D domains with different resolutions. The layout of both model domains can be seen in Figure 5.1. The first domain is the larger of the two and has coarser resolution. This will be referred to as the 'parent' domain and on it we run the 'parent' model. The parent domain is divided up to contain N internal spatial gridpoints and there are $\mathcal{T}$ parent timesteps in the assimilation window.

Within the parent domain is nested a smaller domain, the LAM domain, and on this domain we run the LAM. The LAM covers D parent grid spaces where $D = B2 - B1$ and $B1, B2$ are the parent gridpoints corresponding to the boundaries of the limited area domain. The LAM domain can have a finer resolution than the parent domain. There are M internal spatial gridpoints in the LAM and S LAM timesteps in the assimilation

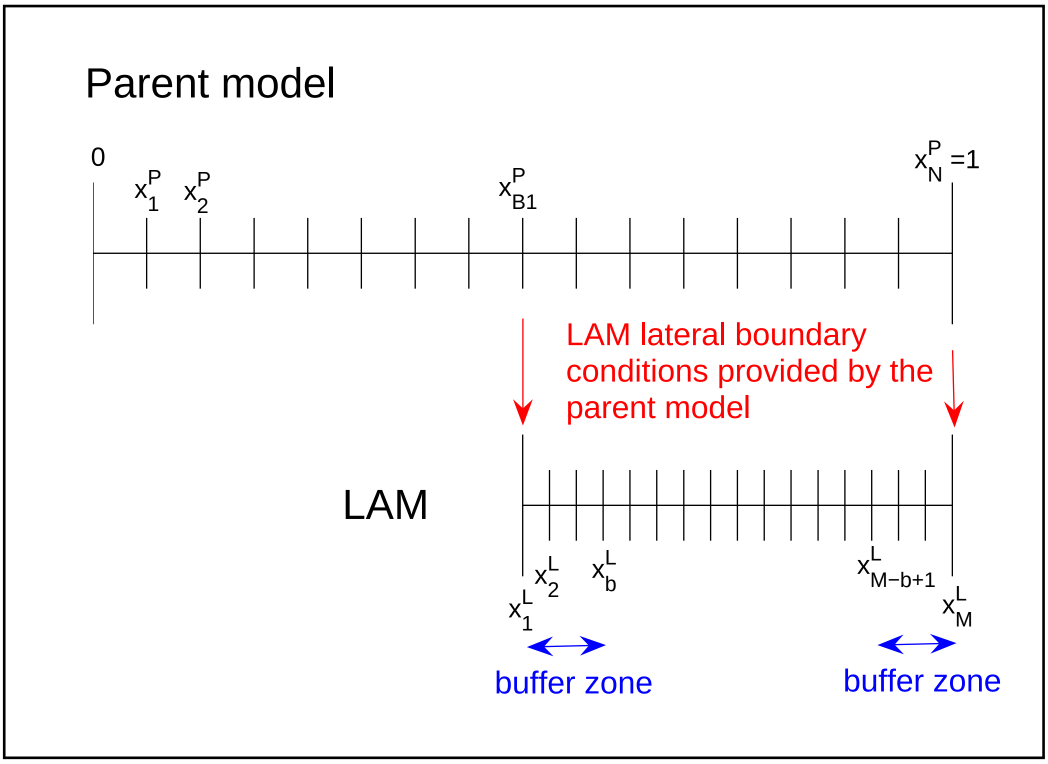


Figure 5.1: Diagram of model domains.

window.

The lateral boundary conditions (LBCs) for the LAM are provided by the parent model. In order to relax the solution on the interior of the LAM domain to the values prescribed at the boundaries there is a buffer zone implemented at the boundaries of the LAM. The buffer zone covers b LAM gridpoints and a Davies Relaxation scheme is used [20], [21]. We use Davies Relaxation because this is what is used operationally [79], as discussed in Section 4.1. We use a linear interpolation function

$$\alpha_i = 1 - [i/b],$$

where b is the width of the buffer zone. When the resolution between the parent and LAM differs, the LBCs provided by the parent model are interpolated onto the resolution of the LAM grid.

5.2 Modifications to the 4D-Var algorithm for the LAM domain

As discussed in Section 2.2, to generate an analysis using 4D-Var we must minimise the cost function (2.17)

$$J(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x^b})^{\mathcal{T}} \boldsymbol{B^{-1}}(\boldsymbol{x} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - \boldsymbol{H_k}(\boldsymbol{x_k}))^{\mathcal{T}} \boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H_k}(\boldsymbol{x_k})),$$

subject to the model equation (2.18)

$$\boldsymbol{x_{k+1}} = \boldsymbol{M_k x_k}.$$

However, the LAM has Davies Relaxation performed in the buffer zone which takes values from the parent model so it is not possible to write the model equations including the modifications at the boundaries in this form. We can however write the LAM model equations as

$$\boldsymbol{x_{k+1}} = \widehat{\boldsymbol{M_k}}\boldsymbol{x_k} + \boldsymbol{P x^p_{k+1}}, \tag{5.1}$$

where $\boldsymbol{x_k}$ is now the state vector on the LAM domain at time level k, $\widehat{\boldsymbol{M_k}}$ is a modified matrix at time level k that takes account of the scaling factor in the buffer zone due to the Davies relaxation, $\boldsymbol{P}$ is a matrix of scaling factors for the influence of the global values in the buffer zone due to the Davies relaxation and $\boldsymbol{x^p_k}$ is a vector of values from the parent model at time level k.

For simplicity we assume that the matrix $\widehat{\boldsymbol{M_k}}$ is the same at all time levels and we can therefore replace $\widehat{\boldsymbol{M_k}}$ in equation (5.1) with the constant matrix $\widehat{\boldsymbol{M}}$. Assuming that $\widehat{\boldsymbol{M}}$ represents a three-point discretisation scheme of the form

$$\boldsymbol{x_{i,k+1}} = \eta \boldsymbol{x_{i-1,k}} + \gamma \boldsymbol{x_{i,k}} + \mu \boldsymbol{x_{i+1,k}},$$

we have

$$\widehat{\boldsymbol{M}} = \left(\begin{array}{cccccccccccc}
0 & 0 & & & & & \dots & & & & & 0 \\
\psi_1\eta & \psi_1\gamma & \psi_1\mu & 0 & & & & \dots & & & & 0 \\
0 & \psi_2\eta & \psi_2\gamma & \psi_2\mu & 0 & & & \dots & & & & 0 \\
 & & & \ddots & & & & & & & & \\
0 & & 0 & \psi_{b-1}\eta & \psi_{b-1}\gamma & \psi_{b-1}\mu & 0 & & \dots & & & 0 \\
0 & & & 0 & \eta & \gamma & \mu & 0 & \dots & & & 0 \\
 & & & & & & \ddots & & & & & \\
0 & & & & & 0 & \eta & \gamma & \mu & 0 & & 0 \\
0 & & & & \dots & 0 & \psi_{b-1}\eta & \psi_{b-1}\gamma & \psi_{b-1}\mu & 0 & & 0 \\
0 & & & & \dots & & 0 & \psi_{b-2}\eta & \psi_{b-2}\gamma & \psi_{b-2}\mu & 0 & 0 \\
 & & & & & & & & & \ddots & & \\
0 & & & & \dots & & & & 0 & \psi_1\eta & \psi_1\gamma & \psi_1\mu \\
0 & & & & \dots & & & & & & & 0
\end{array} \right), \tag{5.2}$$

and

$$\boldsymbol{P} = \left(\begin{array}{cccccccccccc}
\phi_0 & 0 & & & & & \dots & & & & & 0 \\
0 & \phi_1 & 0 & & & & \dots & & & & & 0 \\
0 & 0 & \phi_2 & 0 & & & \dots & & & & & 0 \\
\vdots & & & & & & & & & & & \\
0 & & & 0 & \phi_{b-1} & 0 & & \dots & & & & 0 \\
0 & & & & 0 & 0 & 0 & & & & & 0 \\
 & & & & & \ddots & & & & & & \\
0 & & & & 0 & 0 & 0 & & & & & 0 \\
0 & & & & & \dots & 0 & \phi_{b-1} & 0 & & & 0 \\
0 & & & & & \dots & & 0 & \phi_{b-2} & 0 & & 0 \\
\vdots & & & & & & & & & & & \\
0 & & & & & \dots & & & & 0 & \phi_1 & 0 \\
0 & & & & & \dots & & & & & 0 & \phi_0
\end{array} \right), \tag{5.3}$$

where $\boldsymbol{\phi} = \{\phi_i\}_{i=0}^{b-1}$ is a vector of Davies Relaxation coefficients and $\psi_i = 1 - \phi_i$, where the subscript here refers to the spatial step.

It is clear that this change in the form of the model equations can have no effect on ∇J_b [equation (2.22)] but it is less clear what effect it has on ∇J_o [equation (2.23)]. Therefore, we now consider ∇J_o. From the model equations (5.1) we have

$$
\begin{aligned}
\boldsymbol{x_1} &= \widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}, \\
\boldsymbol{x_2} &= \widehat{\boldsymbol{M}}\boldsymbol{x_1} + \boldsymbol{P}\boldsymbol{x_2^p} = \widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p}, \\
\boldsymbol{x_3} &= \widehat{\boldsymbol{M}}\boldsymbol{x_2} + \boldsymbol{P}\boldsymbol{x_3^p} = \widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p}) + \boldsymbol{P}\boldsymbol{x_3^p}, \\
&\vdots \\
\boldsymbol{x_{\mathcal{T}}} &= \widehat{\boldsymbol{M}}\boldsymbol{x_{\mathcal{T}-1}} + \boldsymbol{P}\boldsymbol{x_{\mathcal{T}}^p} = \widehat{\boldsymbol{M}}(\cdots(\widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p})\cdots) + \boldsymbol{P}\boldsymbol{x_{\mathcal{T}}^p}.
\end{aligned}
$$

Inserting these into J_o from equation (2.17) gives

$$
\begin{aligned}
J_o &= \frac{1}{\mathcal{T}}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - \boldsymbol{H_k}(\boldsymbol{x_k}))^T\boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H_k}(\boldsymbol{x_k})), \\
&= \frac{1}{2}(\boldsymbol{y_0} - \boldsymbol{H_0}(\boldsymbol{x_0}))^T\boldsymbol{R_0^{-1}}(\boldsymbol{y_0} - \boldsymbol{H_0}(\boldsymbol{x_0})) \\
&\quad + \frac{1}{2}(\boldsymbol{y_1} - \boldsymbol{H_1}(\boldsymbol{x_1}))^T\boldsymbol{R_1^{-1}}(\boldsymbol{y_1} - \boldsymbol{H_1}(\boldsymbol{x_1})) \\
&\quad + \frac{1}{2}(\boldsymbol{y_2} - \boldsymbol{H_2}(\boldsymbol{x_2}))^T\boldsymbol{R_2^{-1}}(\boldsymbol{y_2} - \boldsymbol{H_2}(\boldsymbol{x_2})) \\
&\quad + \cdots + \frac{1}{2}(\boldsymbol{y_{\mathcal{T}}} - \boldsymbol{H_{\mathcal{T}}}(\boldsymbol{x_{\mathcal{T}}}))^T\boldsymbol{R_{\mathcal{T}}^{-1}}(\boldsymbol{y_{\mathcal{T}}} - \boldsymbol{H_{\mathcal{T}}}(\boldsymbol{x_{\mathcal{T}}})), \\
&= \frac{1}{2}(\boldsymbol{y_0} - \boldsymbol{H_0}(\boldsymbol{x_0}))^T\boldsymbol{R_0^{-1}}(\boldsymbol{y_0} - \boldsymbol{H_0}(\boldsymbol{x_0})) \\
&\quad + \frac{1}{2}(\boldsymbol{y_1} - \boldsymbol{H_1}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}))^T\boldsymbol{R_1^{-1}}(\boldsymbol{y_1} - \boldsymbol{H_1}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p})) \\
&\quad + \frac{1}{2}(\boldsymbol{y_2} - \boldsymbol{H_2}(\widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p}))^T\boldsymbol{R_2^{-1}} \\
&\qquad\qquad (\boldsymbol{y_2} - \boldsymbol{H_2}(\widehat{\boldsymbol{M}}(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p})), \\
&\quad + \cdots + \frac{1}{2}(\boldsymbol{y_{\mathcal{T}}} - \boldsymbol{H_{\mathcal{T}}}(\widehat{\boldsymbol{M}}(\cdots(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p})\cdots) + \boldsymbol{P}\boldsymbol{x_{\mathcal{T}}^P})^T\boldsymbol{R_{\mathcal{T}}^{-1}} \\
&\qquad\qquad (\boldsymbol{y_{\mathcal{T}}} - \boldsymbol{H_{\mathcal{T}}}(\widehat{\boldsymbol{M}}(\cdots(\widehat{\boldsymbol{M}}\boldsymbol{x_0} + \boldsymbol{P}\boldsymbol{x_1^p}) + \boldsymbol{P}\boldsymbol{x_2^p})\cdots) + \boldsymbol{P}\boldsymbol{x_{\mathcal{T}}^P})^T. \qquad (5.4)
\end{aligned}
$$

Notice here that the $\boldsymbol{P}\boldsymbol{x_k^p}$ terms do not depend on $\boldsymbol{x_0}$, so the gradient with respect to $\boldsymbol{x_0}$ is just an expression like equation (2.23)

$$
\begin{aligned}
-\nabla J_o &= \boldsymbol{H_0^T}\boldsymbol{d_0} + \widehat{\boldsymbol{M}}^{\boldsymbol{T}}[\boldsymbol{H_1^T}\boldsymbol{d_1} + \widehat{\boldsymbol{M}}^{\boldsymbol{T}}[\boldsymbol{H_2^T}\boldsymbol{d_2} + \\
&\qquad \widehat{\boldsymbol{M}}^{\boldsymbol{T}}[\boldsymbol{H_3^T}\boldsymbol{d_3} + \cdots + \widehat{\boldsymbol{M}}^{\boldsymbol{T}}\boldsymbol{H_{\mathcal{T}}^T}\boldsymbol{d_{\mathcal{T}}}]\cdots]], \qquad (5.5)
\end{aligned}
$$

where

$$\boldsymbol{d_k} = \boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H_k}(\boldsymbol{x_k})),$$

but now $\widehat{\boldsymbol{M}}$ is given by equation (5.2). This illustrates that the LAM model can use the same code to generate the gradient of the cost function as the parent model by simply using the matrix $\widehat{\boldsymbol{M}}$ instead of the standard matrix $\boldsymbol{M}$.

5.3 Working in incremental space

The state vector $\boldsymbol{x}$ can be expressed as

$$\boldsymbol{x} = \boldsymbol{x^b} + \boldsymbol{\delta x},$$

where $\boldsymbol{\delta x}$ are known as the increments. If we are using a linear model then modelling the increments is equivalent to modelling the full state and we can easily convert our model to an incremental model. We wish to model the increments instead of the full state in this thesis because it allows us to perform the data assimilation in spectral space as well as in physical (gridpoint) space. This spectral data assimilation scheme is discussed in Section 5.4.

When working with the full model state, the LAM boundaries are forced to remain equal to the values prescribed by the parent model. Since it is this same parent model that provides the background for the data assimilation on the LAM, when working in incremental space the increments on the boundaries are automatically set to zero [i.e. we impose zero boundary conditions on the LAM data assimilation in incremental space]. It is these zero boundary conditions that we utilise for the spectral scheme.

It is worth noting here that since we have constant zero boundary values when working in incremental space, in this work we omit the two boundary gridpoints from our state vector. Therefore, the incremental state vector is $\boldsymbol{\delta x} = \{\delta x_i\}_{i=1}^{i=N-1}$ compared to the full state vector $\boldsymbol{x} = \{x_i\}_{i=0}^{i=N}$. Omitting the boundary points from the state vector does not alter the output generated by the LAM but it simplifies the spectral transform that is to be used for the spectral data assimilation scheme. This will be discussed in Section 5.4

As well as converting our model to incremental space, we can also rewrite the cost function in terms of increments. Starting from equation (2.17) we have

$$\begin{aligned} J(\boldsymbol{x_0}) &= \frac{1}{2}(\boldsymbol{x_0} - \boldsymbol{x^b})^T \boldsymbol{B^{-1}}(\boldsymbol{x_0} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - \boldsymbol{H_k x_k})^T \boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H_k x_k}), \\ &= \frac{1}{2}\boldsymbol{\delta x_0}^T \boldsymbol{B^{-1}} \boldsymbol{\delta x_0} + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{y_k} - \boldsymbol{H_k x_k^b} - \boldsymbol{H_k \delta x_k})^T \boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H_k x_k^b} - \boldsymbol{H_k \delta x_k}), \\ &= \frac{1}{2}\boldsymbol{\delta x_0}^T \boldsymbol{B^{-1}} \boldsymbol{\delta x_0} + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k \delta x_k})^T \boldsymbol{R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k \delta x_k}), \qquad (5.6) \\ &= J(\boldsymbol{\delta x_0}) \end{aligned}$$

where $\boldsymbol{\delta y_k} = \boldsymbol{y_k} - \boldsymbol{H_k x_k^b}$. We also now have the model constraint

$$\boldsymbol{\delta x_{k+1}} = \boldsymbol{M_k \delta x_k}. \tag{5.7}$$

From equation (5.6) it is easy to rewrite the gradients in terms of increments. The gradient of J_b with respect to δx_0 becomes

$$\nabla J_b = \boldsymbol{B^{-1} \delta x_0},$$

and the gradient of J_o with respect to δx_0 is still

$$-\nabla J_o = \boldsymbol{H_0^T d_0} + \boldsymbol{M^T}[\boldsymbol{H_1^T d_1} + \boldsymbol{M_2^T}[\boldsymbol{H_2^T d_2} + \cdots + \boldsymbol{M^T H_{\mathcal{T}}^T d_{\mathcal{T}}}]\cdots],$$

but now

$$\boldsymbol{d_k} = \boldsymbol{R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k \delta x_k}).$$

5.4 A gridpoint and a spectral scheme for the LAM domain

5.4.1 The gridpoint scheme

The 4D-Var algorithm discussed so far performs the data assimilation in gridpoint (physical) space. The algorithm that performs the data assimilation in incremental gridpoint space with control variable $\boldsymbol{\delta x}$, using the cost function (5.6) will be referred to throughout this thesis as the ‘gridpoint’ scheme.

5.4.2 The spectral scheme

As well as the gridpoint version of the algorithm we also set up a 4D-Var algorithm that performs the data assimilation in spectral space via a sine transform.

The sine transform
For a periodic function $f = f_j$, defined on a given domain divided into equally spaced

discrete intervals with grid points numbered $j = 0, 1, 2, \cdots, N$ *and* $f_0 = f_N = 0$*, the Fourier sine transform is defined to be*

$$\textit{sine transform}(f_j) = \widetilde{f_\kappa} = \sum_{j=1}^{N-1} f_j \sin(\pi j \kappa/(N)), \tag{5.8}$$

where κ *is the wavenumber [68].*

The sine transform and its properties are discussed in Section 6.4.

The spectral version of the algorithm will be referred to throughout this thesis as the ‘spectral’ scheme. This spectral version is developed because it is a spectral scheme that is used operationally on the LAM by the Met Office, as discussed in Section 4.2.1.

Both the gridpoint and the spectral schemes are used in this thesis. By using both a gridpoint and a spectral based data assimilation scheme we can compare the output of the two methods when given identical data and investigate the effect of the sine transform.

To change the data assimilation scheme to a spectral one we need to perform a control variable transform, as discussed in Section 2.3.4. The control variable transform used here is the sine transform and it transforms the increments from physical to spectral space. We define control variable transform

$$\boldsymbol{z} = \boldsymbol{W}\boldsymbol{x}, \tag{5.9}$$

and an inverse transform

$$\boldsymbol{x} = \boldsymbol{U}\boldsymbol{z}, \tag{5.10}$$

where $\boldsymbol{W} \in \mathbb{R}^{N \times N}$ is the sine transform and $\boldsymbol{U} \in \mathbb{R}^{N \times N}$ is the inverse sine transform.

In matrix notation the sine transform $\boldsymbol{W}$ can be written as

$$\boldsymbol{W} = \{W_{j\kappa}\}, \quad j = 1, \ldots, N-1 \quad \kappa = 1, \ldots, N-1 \tag{5.11}$$

where

$$W_{j\kappa} = \sqrt{\frac{2}{N}} \sin(\pi j \kappa/(N)). \tag{5.12}$$

The $\sqrt{\frac{2}{N}}$ term is a normalisation factor. Matrix $\boldsymbol{W}$ has a similar form to the more usual Fourier matrix [80]. The dimensions of the matrix $\boldsymbol{W}$ here fit that of the incremental state vector $\boldsymbol{\delta x}$, where the boundary points have been omitted.

Now that we have our new control variable we consider the effect it has on the cost function. We start from equation (5.6) but now in terms of $\boldsymbol{z}$ instead of $\boldsymbol{x}$.

$$J(\boldsymbol{z_0}) = \frac{1}{2}(\boldsymbol{U\delta z_0})^T\boldsymbol{B^{-1}}(\boldsymbol{U\delta z_0}) + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k U \delta z_k})^T\boldsymbol{R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k U \delta z_k}).$$

This can be written as

$$J(\boldsymbol{z_0}) = \frac{1}{2}\boldsymbol{\delta z_0}^T\boldsymbol{\Sigma^{-1}}\boldsymbol{\delta z_0} + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k U \delta z_k})^T\boldsymbol{R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k U \delta z_k}), \qquad (5.13)$$

where

$$\boldsymbol{\Sigma^{-1}} = \boldsymbol{U^T B^{-1} U}. \qquad (5.14)$$

We now consider the effect the control variable transform has on the gradient of the cost function. The gradient of J_b with respect to the new control variable is

$$\nabla J_b(\boldsymbol{z_0}) = \boldsymbol{\Sigma^{-1}\delta z_0}, \qquad (5.15)$$

and the gradient of J_o with respect to the new control variable is

$$-\nabla J_o(\boldsymbol{z_0}) = \sum_{k=0}^{\mathcal{T}}\boldsymbol{U^T}\underbrace{\widehat{\boldsymbol{M}}^{\boldsymbol{T}}\widehat{\boldsymbol{M}}^{\boldsymbol{T}}\cdots\widehat{\boldsymbol{M}}^{\boldsymbol{T}}}_{\text{k functions}}\boldsymbol{H_k^T R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{H_k U \delta z_k}). \qquad (5.16)$$

As we see, the only effect is the addition of a $\boldsymbol{U}^T$ term at the start of the ∇J_o equation and the change from $\boldsymbol{B^{-1}}$ to $\boldsymbol{\Sigma^{-1}}$ in the ∇J_b equation. These changes are easily made to the gridpoint data assimilation code.

5.5 Summary

We have introduced a general, nested LAM domain. We have discussed the use of $\boldsymbol{M}^T$ as the adjoint model in the 4D-Var algorithm and also the modifications to $\boldsymbol{M}$ that are required to apply this to the LAM domain. A gridpoint and a spectral scheme for the LAM domain have also been discussed. We now go on to consider the implications the LAM domain has on the DFT methods that were introduced in Chapter 3. We also compare the results from the DFT with those generated using a Fourier sine transform.

Chapter 6

The discrete Fourier transform on a LAM domain

In Chapter 3 we demonstrated methods of using the discrete Fourier transform (DFT) and its power spectrum to understand properties of a solution, in particular, how we can use the power spectrum to identify the wavenumber and amplitude of the components of a solution. However, there can be a problem when we try to apply the DFT over the LAM domain: As discussed in Chapter 4, we may have waves whose wavelengths are longer than the LAM domain. These wavenumbers are not part of the LAM spectrum. In this chapter we aim to understand where the information from these waves is projected when we perform the DFT over the LAM domain.

We begin by considering how the DFT is calculated from its definition.

6.1 Calculating the DFT from its definition

We consider the DFT performed on a general sine wave of amplitude α and wavenumber θ

$$\varphi_j = \alpha \sin(2\pi\theta x_j), \tag{6.1}$$

and a general cosine wave of amplitude β and wavenumber ϑ

$$\phi_j = \beta \cos(2\pi\vartheta x_j), \tag{6.2}$$

where $\theta \in \mathbb{R}$, $\vartheta \in \mathbb{R}$, $x_j = j\Delta x$, $j = 0, 1, 2, \cdots, N-1$, $\Delta x = L/N$ is the gridspacing and L is the length of the domain (i.e. $x \in [0, L)$).

If we further restrict θ and ϑ such that $\theta L \in \mathbb{N}$ and $\vartheta L \in \mathbb{N}$, then following the same methods used in Section 3.2.1, the orthogonality relation defined by Lemma 3.1 and also utilising the identity

$$\cos(x) \equiv \frac{1}{2}\left(e^{ix} + e^{-ix}\right), \tag{6.3}$$

the DFTs of φ_j and ϕ_j can be written as

$$\widehat{\varphi}_k = \begin{cases} -i\alpha N/2 & \text{if } k - \theta L = 0 \text{ (i.e. } k = \theta L), \\ i\alpha N/2 & \text{if } k + \theta L = N \text{ (i.e. } k = N - \theta L), \\ 0 & \text{otherwise,} \end{cases} \tag{6.4}$$

and

$$\widehat{\phi}_k = \begin{cases} \beta N/2 & \text{if } k - \vartheta L = 0 \text{ (i.e. } k = \vartheta L), \\ \beta N/2 & \text{if } k + \vartheta L = N \text{ (i.e. } k = N - \vartheta L), \\ 0 & \text{otherwise.} \end{cases} \tag{6.5}$$

Since the DFT is a linear operator, for any function f_j which is a linear combination of sine and cosine waves we can construct the DFT from a combination of these general cases.

In equations (6.4), and (6.5) we assume that $\theta L \in \mathbb{N}$ (or $\vartheta L \in \mathbb{N}$). However, if this assumption does not hold then we have not satisfied the hypothesis of Lemma 3.1 that k has to be an integer, and therefore we cannot use Lemma 3.1 to simplify the DFT. This causes problems on the LAM domain.

The situation $\theta L \notin \mathbb{N}$ can occur on the LAM domain in one of two ways:

1. The wavelength of the wave is longer than the domain of the model ($\theta L < 1$),
2. The wavelength of the wave is shorter than the domain of the model but the domain size is not an interger multiple of the wavelength ($\theta L > 1$).

In this Chapter we will only consider the first case $\theta L < 1$ (the long-wave case).

Before we consider the DFT over the LAM domain for a long-wave, we need first to understand what the wavenumbers we have on the parent domain correspond to on a

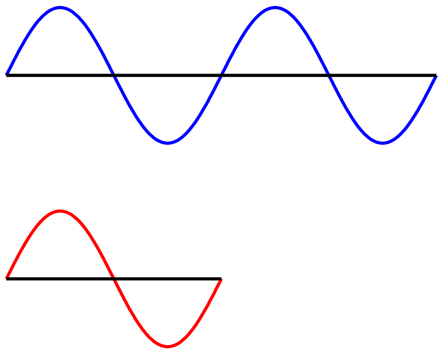


Figure 6.1: Comparing the same wave on domains of different length.

LAM domain. For simplicity we take $x_p \in (0, 1]$ as our parent domain, i.e. $L = 1$ for the parent domain, and therefore for the LAM domain $L < 1$.

6.2 Wavelengths on a LAM domain

On the parent domain $x_p \in (0, 1]$ and we have the sine wave with wavenumber κ_p

$$u = \sin(2\pi\kappa_p x_p). \tag{6.6}$$

For a LAM domain $x \in (0, L]$ where $L < 1$, only a section of the wave u will fit in the LAM domain.

To find the wavenumber of this section of wave on the LAM domain, we rewrite u as

$$u = \sin\left(\frac{2\pi\kappa x}{L}\right), \tag{6.7}$$

where κ is to be found.

We equate the arguments of equations (6.6) and (6.7) to find

$$\kappa = L\kappa_p,$$

i.e. wavenumbers on the LAM domain $x \in (0, L]$, where $L = 1/m$, are $1/m$ what they are on the parent domain $x_p \in (0, 1]$.

This is illustrated in Figure 6.1 for the case of a LAM domain which is half that of the parent (i.e. $m = 2$). As can be seen, the sine wave on the parent domain has wavenumber 2 (two full wavelengths fit in the parent domain) but the same wave on the LAM domain has wavenumber 1 (only one wavelength fits in the LAM domain).

Now that we understand the equivalence between wavenumbers on the domain $L = 1$ and those on the domain $L < 1$ of the same wave, we can consider the DFT when $L < 1$, knowing where we expect the amplitude to appear in the power spectrum.

6.3 The DFT over a LAM domain

To investigate how the DFT behaves over a LAM domain, and in particular how a 'long wave' is treated by the DFT, we consider the DFT of a function f_j which is a linear combination of sine waves.

We take the example

$$\begin{aligned} f_j &= 2\sin(2\pi jL/N) + 2\sin(4\pi jL/N) \\ &\quad + \sin(8\pi jL/N) + \sin(16\pi jL/N), \end{aligned} \tag{6.8}$$

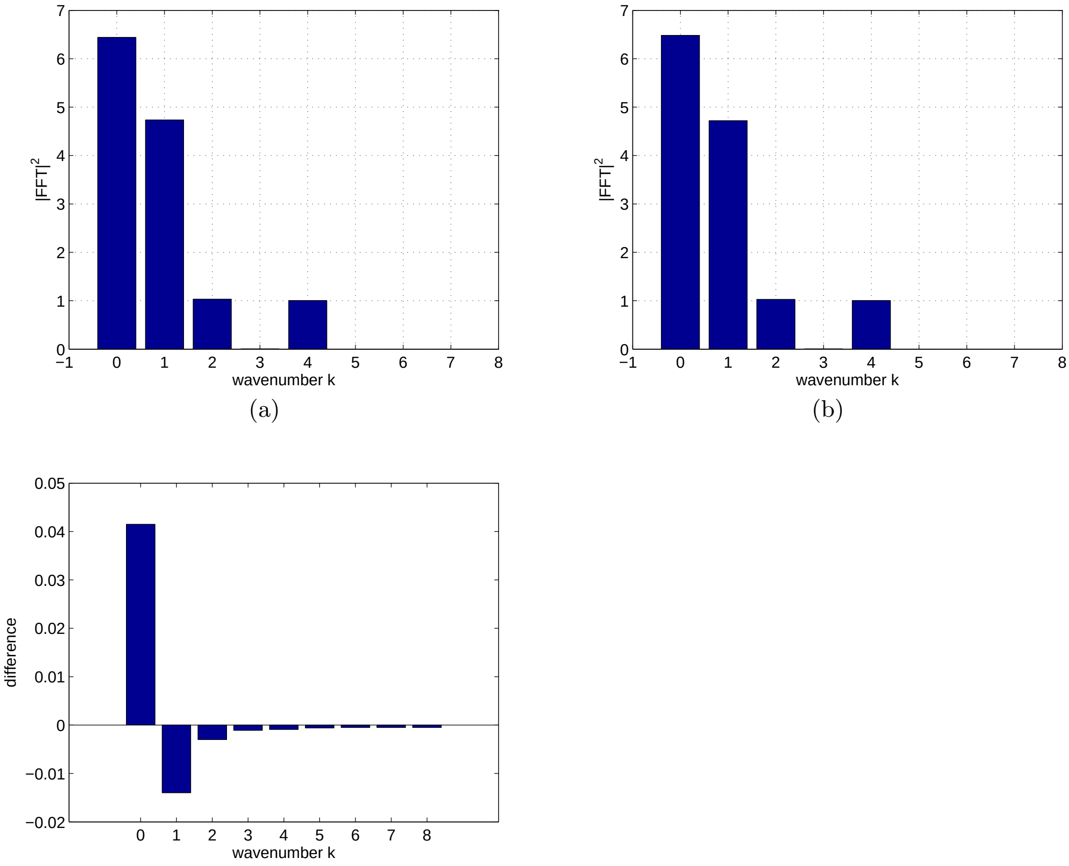
and begin with the LAM domain $L = 1/2$.

6.3.1 $L = 1/2$

Substituting $L = 1/2$ into equation (6.8) gives us our wave as

$$\begin{aligned} f_j &= 2\sin(\pi j/N) + 2\sin(2\pi j/N) \\ &\quad + \sin(4\pi j/N) + \sin(8\pi j/N), \end{aligned} \tag{6.9}$$

on the LAM domain. Here the longest wavelength contained in f_j is longer than L. The power spectrum of f_j, taking $N = 16$, is shown in Figure 6.2(a). As discussed in Section 3.2.2, when plotting the power spectrum all of the DFT coefficients are scaled by the factor $2/N$. Also, in all the power spectra of DFTs shown in the rest of this thesis we only plot the first half of the spectrum, we do not plot the amplitude caused



(a)

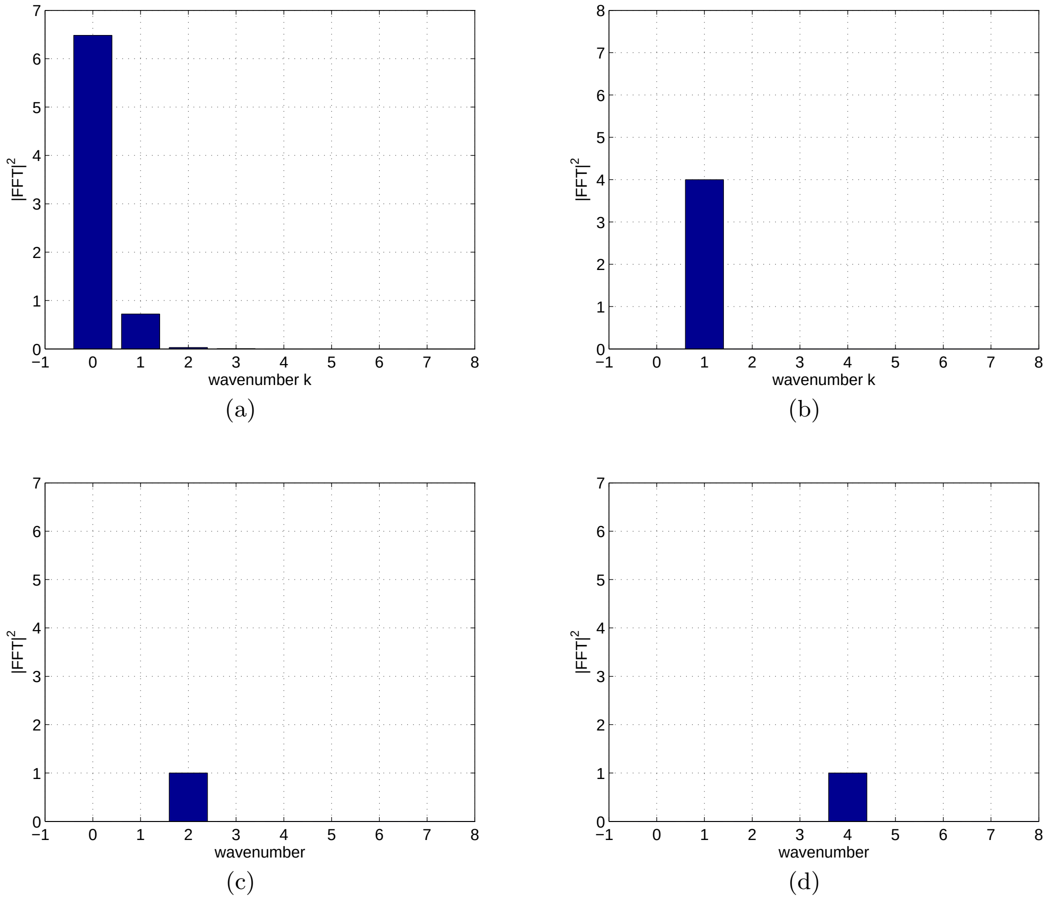
(b)

(c)

Figure 6.2: (a) Power spectrum of f_j with $N = 16$, scaled by the factor $2/N$. (b) Power spectrum of f_j with $N = 256$, scaled by the factor $2/N$. (c) Difference between $N = 256$ and $N = 16$ power spectra amplitudes.

by the symmetry of the complex conjugate, as discussed in Section 3.4.1, i.e. we plot the amplitude at wavenumber κ but not the corresponding amplitude at wavenumber $N - \kappa$.

From equation (6.4) we expect the $2 \sin(2\pi j/N)$ wave to produce a peak with amplitude 4 at $k = 1$, the $\sin(4\pi j/N)$ wave to produce a peak with amplitude 1 at $k = 2$ and the $\sin(8\pi j/N)$ wave to produce a peak with amplitude 1 at $k = 4$. However, we do not know where information about the $2\sin(\pi j/N)$ wave will be sent. It has wavenumber $k = 1/2$ but equation (6.4) requires $k \in \mathbb{N}$. As can be seen in Figure 6.2(a) there is considerable amplitude at $k = 0$. There is also a discrepancy in the amplitude we expect at other wavenumbers. This discrepancy could be a result of errors caused by resolution. With $N = 16$ the short wavelengths may not be well resolved and this error would then show up in the power spectrum. To check if this is the case we re-run the



(a)

(b)

(c)

(d)

Figure 6.3: Power spectra of the individual waves contained in f_j, scaled by the factor $2/N$. (a) $2\sin(\pi j/N)$, (b) $2\sin(2\pi j/N)$, (c) $\sin(4\pi j/N)$ and (d) $\sin(8\pi j/N)$.

DFT calculation using $N = 256$, a high enough resolution to accurately represent all the wavelengths. The power spectrum with $N = 256$ is shown in Figure 6.2(b). Comparison with Figure 6.2(a) shows that it does not appear that changing the resolution has affected the power distribution. This is confirmed by Figure 6.2(c) which plots the difference between the power spectrum with $N = 256$ and the power spectrum with $N = 16$. As can be seen the change in amplitude is at most $\mathrm{O}(10^{-2})$. This suggests the unexpected features we see in the power spectrum are caused by the long wavelength. To investigate this further we consider the wavelengths contained in f_j individually. The power spectra of each wave are shown in Figure 6.3. As can be seen, the waves with wavenumbers $k = 1, 2, 4$ do produce the power spectra we expect. Figure 6.3(a) shows that all the amplitude at $k = 0$ and the changes at $k = 1, 2, 3, \ldots$ result from the long wave.

To understand better the power spectrum of f_j and in particular the effect of the

$2\sin(\pi j/N)$ wave we consider the definition of the DFT in more detail.

Since f_j is a linear combination of sine waves, the DFT of f_j can be written as

$$\widehat{f_k} = a_k + b_k + c_k + d_k, \tag{6.10}$$

where

$$\begin{aligned}
a_k &= \frac{1}{i}\sum_{j=0}^{N-1}\left(e^{-i2\pi j(k-L)/N} - e^{-i2\pi j(k+L)/N}\right),\\
b_k &= \frac{1}{i}\sum_{j=0}^{N-1}\left(e^{-i2\pi j(k-2L)/N} - e^{-i2\pi j(k+2L)/N}\right),\\
c_k &= \frac{1}{2i}\sum_{j=0}^{N-1}\left(e^{-i2\pi j(k-4L)/N} - e^{-i2\pi j(k+4L)/N}\right),\\
d_k &= \frac{1}{2i}\sum_{j=0}^{N-1}\left(e^{-i2\pi j(k-8L)/N} - e^{-i2\pi j(k+8L)/N}\right),
\end{aligned}$$

We first consider $k = 0$. By substituting $k = 0$ and $L = 1/2$ into equation (6.10) we get $b_0 = c_0 = d_0 = 0$ from Lemma 3.1. However a_0 is not in the form of the orthogonality relation so it requires different treatment. Using angle summation formulas and the properties of telescopic sums, a_0 can be written as

$$\begin{aligned}
a_0 &= \frac{1}{i}\sum_{j=0}^{N-1}\left(e^{i\pi j/N} - e^{-i\pi j/N}\right),\\
&= 2\sum_{j=0}^{N-1}\sin(\pi j/N),\\
&= \frac{1}{\sin(\pi/N)}\sum_{j=0}^{N-1}2\sin(\pi j/N)\sin(\pi/N),\\
&= \frac{1}{\sin(\pi/N)}\sum_{j=0}^{N-1}\left[\cos(\pi j/N - \pi/N) - \cos(\pi j/N + \pi/N)\right],\\
&= \frac{-1}{\sin(\pi/N)}\sum_{j=0}^{N-1}\left[\cos((j+1)\pi/N) - \cos((j-1)\pi/N)\right],\\
&= \frac{1}{\sin(\pi/N)}\left[2 + \cos(\pi/N) - \cos((N-1)\pi/N)\right],\\
&\neq 0.
\end{aligned}$$

So at $k = 0$ we have

$$\widehat{f_0} = a_0 = \frac{1}{\sin(\pi/N)}\left[2 + \cos(\pi/N) - \cos((N-1)\pi/N)\right]. \tag{6.11}$$

For the case with resolution $N = 256$, this gives

$$a_0 = \frac{1}{\sin(\pi/256)} \left[2 + \cos(\pi/256) - \cos(255\pi/256)\right]$$

which would give a peak on the power spectrum at $k = 0$ of amplitude corresponding to the value we see at $k = 0$ in Figure 6.3(a) and Figure 6.2(b).

Next we consider $k = 1$. By substituting $k = 1$ and $L = 1/2$ into equation (6.10) we again get $c_1 = d_1 = 0$ from Lemma 3.1 and also get that $b_1 = -iN$. Again however, a_1 cannot be simplified using the orthogonality relation and is left as

$$a_1 = \frac{1}{i} \sum_{j=0}^{N-1} \left(e^{-i\pi j/N} - e^{-i3\pi j/N}\right).$$

So at $k = 1$ we have

$$\widehat{f_1} = a_1 + b_1. \tag{6.12}$$

Again for the case with resolution $N = 256$, this gives

$$a_1 = \frac{1}{i} \sum_{j=0}^{255} \left(e^{-i\pi j/256} - e^{-i3\pi j/256}\right)$$

which would give an amplitude in the power spectrum at $k = 1$ corresponding to the value we see at $k = 1$ in Figure 6.3(a). Also

$$a_1 + b_1 = \frac{1}{i} \sum_{j=0}^{255} \left(e^{-i\pi j/256} - e^{-i3\pi j/256}\right) - i256$$

which would give an amplitude in the power spectrum at $k = 1$ corresponding to the value we see at $k = 1$ in Figure 6.2(b).

If we consider $k = 2$ in the same way we get that $b_2 = d_2 = 0$, $c_2 = -iN/2$ and

$$a_2 = \frac{1}{i} \sum_{j=0}^{N-1} \left(e^{-i3\pi j/N} - e^{-i5\pi j/N}\right).$$

So at $k = 2$ we have

$$\widehat{f_2} = a_2 + c_2. \tag{6.13}$$

For the case with resolution $N = 256$, this gives

$$a_2 = \frac{1}{i} \sum_{j=0}^{255} \left(e^{-i3\pi j/256} - e^{-i5\pi j/256}\right),$$

N	DFT	amplitude in power spectrum
8	10.0547	6.3185
16	20.3063	6.4429
32	40.7109	6.4741
128	162.9665	6.4839
256	325.9452	6.4844

Table 6.1: Value of $\widehat{f_0}$ for different values of N.

which would give an amplitude in the power spectrum at $k = 2$ corresponding to the value we see at $k = 2$ in Figure 6.3(a). Also

$$a_2 + c_2 = \frac{1}{i}\sum_{j=0}^{255}\left(e^{-i3\pi j/256} - e^{-i5\pi j/256}\right) - i128,$$

which would give an amplitude in the power spectrum at $k = 2$ corresponding to the value we see at $k = 2$ in Figure 6.2(b).

Continuing with these calculations shows that d_k will give us a peak in the power spectrum at $k = 4$ and that all other non-zero values are contributed by a_k, but that the magnitude of these contributions quickly decays for $k > 1$.

As we have seen, the majority of the long wave information is being sent to $k = 0$, but it also appears that the exact amplitude at $k = 0$ may also depend on the number of gridpoints in the domain. To investigate this we again consider the DFT of f_j over the domain $L = 1/2$ but now we vary the number of gridpoints. From equation (6.11) we have, at $k = 0$,

$$\widehat{f_0} = \frac{1}{\sin(\pi/N)}\left[2 + \cos(\pi/N) - \cos((N-1)\pi/N)\right]. \tag{6.14}$$

The value of $\widehat{f_0}$ for different values of N is shown in Table 6.1. From it we see that the value of $\widehat{f_0}$ is dependent on the value of N but as the value of N increases the value of $\widehat{f_0}$ approaches a limit. This is because for large N, $\pi/N << 1$ and for $x << 1$ we have $\sin(x) \approx x$ so the coefficient in equation (6.14), $1/\sin(\pi/N)$, behaves as N/π for large N. Since the DFT has a normalisation factor of $2/N$ this coefficient becomes simply $2/\pi$, which is independent of N. Therefore for large N (high resolution) the amplitude at $k = 0$ does not vary with N. However, for small values of N the coefficient in equation (6.14) cannot be approximated by a constant and the amplitude at $k = 0$

does vary as N increases. For very low resolutions this variation could be significant, increasing N from 8 to 16 will increase the amplitude at $k = 0$ by $O(10^{-1})$, but as the resolution increases the amplitude at $k = 0$ approaches a limit.

To investigate whether the majority of the long-wave information is always sent to $k = 0$ we now consider the definition of the DFT for a general LAM domain of length L.

6.3.2 A general case

The continuous Fourier series approximation of a function has many known properties, including convergence results for specific classes of functions. The discrete Fourier series interpolation has convergence properties very similar to those of the continuous case. Of particular interest here is the fact that the discrete case shares the same asymptotic behaviour as the continuous case [37, p.27].

Following the notation used in [37, Section 2], for a continous function $f(\xi)$, on an equidistant grid with an even number $2N$ of gridpoints $\xi_j \in [0, 2\pi\}$, where

$$\xi_j = 2\pi j/2N,$$

the discrete Fourier coefficients $\widehat{f_k}$ are given by[1]

$$\widehat{f_k} = \frac{1}{2N\widehat{c_k}} \sum_{j=0}^{2N-1} f(\xi_j) e^{-ik\xi_j},$$

where

$$\widehat{c_k} = \begin{cases} 2 & \text{if } |k| = N/2, \\ 1 & \text{if } |k| < N/2. \end{cases}$$

This defines a new projection of f

$$\mathcal{I}_{2N} f(\xi) = \sum_{|n|<N} \widehat{f_k} e^{ik\xi}.$$

This is the discrete Fourier transform, based on an even number of gridpoints. $\mathcal{I}_{2N}$ is also the interpolation operator.

Using this notation, the error estimate for the discrete approximation can now be stated as

[1] This is essentially equivalent to the definition given in Section 3.1 but we now use $2N$ gridpoints instead of N.

Theorem 6.1 *For any $f(\xi) \in W_p^r[0,2\pi]$ with $r > 1/2$, there exists a positive constant C, independent of N, such that*

$$\|f - \mathcal{I}_{2N} f\|_{L^2[0,2\pi]} \leq CN^{-r} \left\| f^{(r)} \right\|_{L^2[0,2\pi]} .$$

where $W_p^q[0,2\pi]$ is the Sobolev space, defined by the norm

$$\|f\|_{W_p^q[0,2\pi]} = \left(\sum_{|k| \leq \infty} (1 + k^{2q}) \left| \widehat{f}_k \right|^2 \right)^{1/2} .$$

Theorem 6.1 tells us that the rate of convergence of the discrete expansion depends only on the smoothness of the function being approximated [37, p.41].

In particular, a function $f(\xi) \in C_p^\infty[0,2\pi]$ has a convergence rate faster than any algebraic order of N^{-1}. This is known as spectral accuracy.

Theorem 6.1 also tells us that the majority of long-wave information will be sent to wavenumber $k = 0$. If we consider the wave (6.9) used in Section 6.3.1 for example, the derivative does not have a continuous periodic extension. The rate of convergence will therefore only be $1/N$. However, a convergence rate of $1/N$ still means the majority of the 'long-wave' information is sent to wavenumber $k = 0$. This will be the case for a single 'long-wave' or a linear combination of several.

If, as in Chapter 3, we consider the sine and cosine waves as a basis for function space, then for a function f there will be a contribution at $k = 0$ from every component of f with a wavelength longer than the domain L. [2]

This general result would suggest that it is important to treat well wavenumber $k = 0$ if we want to accurately represent long waves in a numerical model.

This allocation of the long-wave information to wavenumber $k = 0$ could cause a problem if instead of performing a DFT we perform a sine transform, as a sine transform will

[2] It is worth noting here that this combining of contributions may lead to a reduction in the amplitude in the power spectrum rather than an increase. This is due to the fact that some of the contributions may have a negative sign, and they are combined and then squared to produce the power spectrum. Therefore, although the DFT itself is linear, the power spectrum of a function made up of two waves is not necessarily the power spectrum of each separate wave added together. This is clearly demonstrated in Figures (6.2(a)) and (6.3).

always have zero amplitude at $k = 0$. To investigate this we now perform a sine transform on the function f_j from equation (6.8).

6.4 The sine transform

From Section 5.4.2, for a periodic function $f = f_j$, defined on a given domain divided into equally spaced discrete intervals with grid points numbered $j = 0, 1, 2, \cdots, N$ and $f_0 = f_N = 0$, the Fourier sine transform is defined to be

$$\text{sine transform}(f_j) = \widetilde{f}_k = \sqrt{\frac{2}{N}} \sum_{j=1}^{N-1} f_j \sin(\pi jk/(N)),$$

where k is the wavenumber.

For large values of N, the sine transform can be calculated utilising efficient code available for the FFT. However, due to the (small) size of the vectors being considered here, we can simply perform the sine transform via matrix multiplication, as described in Section 5.4.2.

To understand better the sine transform we start by considering its calculation from the definition.

6.4.1 Calculating the sine transform from its definition

We begin by considering the effect of the sine transform on a general sine wave, as we did for the DFT in Section 3.2.1. We have a general sine wave with wavenumber θ and amplitude α

$$\varphi_j = \alpha \sin(2\pi\theta x_j).$$

The sine transform of φ_j over the domain $[0, L)$ is

$$\begin{aligned} \widetilde{\varphi}_k &= \sum_{j=1}^{N-1} \varphi_j \sin(\pi jk/N), \\ &= \alpha \sum_{j=1}^{N-1} \sin(2\pi\theta jL/N) \sin(\pi jk/N). \end{aligned}$$

Using the identity

$$\sin(x) \equiv \frac{1}{2i}\left(e^{ix} - e^{-ix}\right), \tag{6.15}$$

and that $\sin(0) = 0$, the sine transform of φ_j becomes

$$\widetilde{\varphi}_k = \frac{-\alpha}{4}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi j(2\theta L+k)}{N}} - e^{\frac{i\pi j(2\theta L-k)}{N}} - e^{\frac{-i\pi j(2\theta L-k)}{N}} + e^{\frac{-i\pi j(2\theta L+k)}{N}}\right]. \tag{6.16}$$

Equation (6.16) can be simplified for some cases using Lemma 3.1.

As with the DFT, the sine transform is a linear function, so now if we consider the sine and cosine waves as a basis for function space, we can construct the sine transform for any function f_j from a combination of the general cases.

6.4.2 The sine transform for a linear combination of sine waves

To compare the sine transform with the DFT we now again consider the case when

$$\begin{aligned} f_j &= 2\sin(2\pi jL/N)) + 2\sin(4\pi jL/N) \\ &\quad + \sin(8\pi jL/N) + \sin(16\pi jL/N), \end{aligned} \tag{6.17}$$

as before.

Using equation (6.16) the sine transform of f_j can be written as

$$\widetilde{f}_k = \overline{a} + \overline{b} + \overline{c} + \overline{d}, \tag{6.18}$$

where

$$\begin{aligned} \overline{a} &= \frac{-1}{2}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi j(2L+k)}{N}} - e^{\frac{i\pi j(2L-k)}{N}} - e^{\frac{-i\pi j(2L-k)}{N}} + e^{\frac{-i\pi j(2L+k)}{N}}\right], \\ \overline{b} &= \frac{-1}{2}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi j(4L+k)}{N}} - e^{\frac{i\pi j(4L-k)}{N}} - e^{\frac{-i\pi j(4L-k)}{N}} + e^{\frac{-i\pi j(4L+k)}{N}}\right], \\ \overline{c} &= \frac{-1}{4}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi j(8L+k)}{N}} - e^{\frac{i\pi j(8L-k)}{N}} - e^{\frac{-i\pi j(8L-k)}{N}} + e^{\frac{-i\pi j(8L+k)}{N}}\right], \\ \overline{d} &= \frac{-1}{4}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi j(16L+k)}{N}} - e^{\frac{i\pi j(16L-k)}{N}} - e^{\frac{-i\pi j(16L-k)}{N}} + e^{\frac{-i\pi j(16L+k)}{N}}\right]. \end{aligned}$$

From equation (6.18) it is easy to see that for any domain size, $\widetilde{f}_k$ is always zero at $k = 0$, as we would expect. Therefore, we need only consider $k \geq 1$.

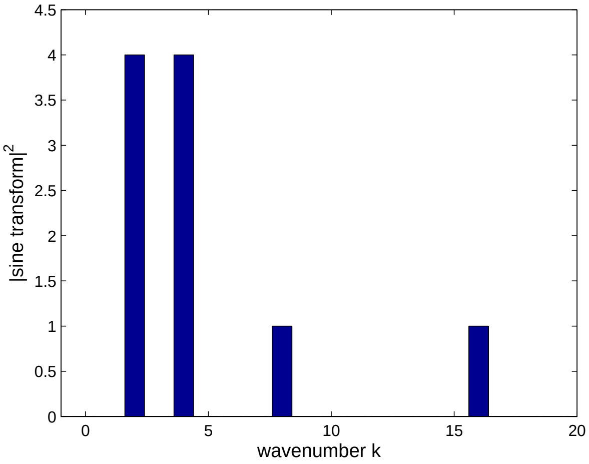


Figure 6.4: The power spectrum of the sine transform of f_j over the domain $L = 1$, scaled by a factor of $2/N$.

As with the DFT, we start by considering the domain $L = 1$. This allows us to compare the power spectrum of the sine transform with that which we would expect with the DFT.

6.4.3 The sine transform when $L = 1$

Figure 6.4 shows the power spectrum of $\widetilde{f_k}$ over the domain $L = 1$ with $N = 256$. As we did for the power spectrum of the DFT, Section 3.2.2, when we plot the power spectrum of the sine transform we apply a scaling factor of $2/N$ to the coefficients which results in an amplitude of one in the power spectrum for a standard trigonometric wave.

As can be seen we get four peaks as we would expect. However, f_j is a combination of waves with wavenumbers $\kappa = 1, 2, 4, 8$ so for the DFT we would expect peaks in the power spectrum at $k = 1, 2, 4, 8$ but the peaks in the power spectrum shown in Figure 6.4 are instead at $k = 2, 4, 8, 16$. To investigate the reason for this apparent shift and also to understand better the sine transform itself we consider its definition.

We first consider the sine transform at $k = 1$. Substituting $k = 1$ and $L = 1$ into equation (6.18) gives $a = b = c = d = 0$, from Lemma 3.1, so $\widetilde{f_1} = 0$ which corresponds to the amplitude we see at $k = 1$ in Figure 6.4.

Considering $k = 2$ in the same way we get $b = c = d = 0$ but now we have

$$\overline{a} = \sum_{j=0}^{N-1} 1 = N,$$

so $\widetilde{f}_2 = N$. After applying the scaling factor $2/N$, this corresponds to the amplitude we see at $k = 2$ in Figure 6.4.

Considering $k = 3$ in the same way we get $a = b = c = d = 0$ so $\widetilde{f}_3 = 0$ which corresponds to the amplitude we see at $k = 3$ in Figure 6.4.

Next considering $k = 4$ in the same way we get $a = c = d = 0$ but now we have

$$\overline{b} = \sum_{j=0}^{N-1} 1 = N,$$

so $\widetilde{f}_4 = N$, which after applying the scaling factor $2/N$, corresponds to the amplitude we see at $k = 4$ in Figure 6.4.

Continuing these calculations, we get

$$\widetilde{f}_k = \begin{cases} N & \text{if } k = 2, \\ N & \text{if } k = 4, \\ N/2 & \text{if } k = 8, \\ N/2 & \text{if } k = 16, \\ 0 & \text{otherwise,} \end{cases}$$

which, having applied the scaling factor $2/N$, give peaks in the power spectrum of 4 at $k = 2, 4$ and of 1 at $k = 8, 16$. These correspond exactly to the peaks we observe in Figure 6.4. The reason for the apparent shift in location of amplitude in Figure 6.4 can also be explained from the definition of the sine transform. If we consider equation (5.8)

$$\widetilde{f}_k = \sum_{j=1}^{N-1} f_j \sin(\pi j k / N),$$

we see that the sine transform does not contain the factor of 2 in the sine term that appears in the exponential term of the DFT. This has the effect of multiplying L by 2 in the definitions of $\overline{a}$, $\overline{b}$, $\overline{c}$, $\overline{d}$ in equation (6.18). This factor of 2 results in a wave with wavenumber κ producing a peak in the power spectrum at $k = 2\kappa$.

Now that we understand the sine transform over a domain large enough to contain full wavelengths of the waves present in f_j, we consider the sine transform over a LAM domain.

Due to the doubling effect on the wavenumbers caused by the sine transform, unlike with the DFT, all the wavenumbers contained by f_j correspond to whole wavenumbers in the power spectrum when $L = 1/2$. Therefore, we instead consider $L = 1/4$.

6.4.4 The sine transform when $L = 1/4$

Before we perform the sine transform we need to know what we expect. From Section 6.3 we know that the wavenumbers in f_j on the domain $L = 1/4$ are $\kappa = 1/4, 1/2, 1, 2$ and from Section 6.4 we know that the sine transform produces peaks in the power spectrum at $k = 2\kappa$. We therefore expect peaks in the power spectrum at $k = 1/2, 1, 2, 4$. However, amplitude cannot be assigned to wavenumbers that are not whole numbers. The $2\sin(2\pi jL/N)$ wave in f_j is too long to be represented and we are therefore unclear as to how this wave will be treated by the sine transform.

The power spectrum of $\widetilde{f}_k$ over the domain $L = 1/4$, with $N = 64$, is shown in Figure 6.5. As can be seen, there is considerable amplitude at $k = 1$. However, there is lower amplitude than we would expect at $k = 2$. Since the $2\sin(2\pi jL/N)$ wave in f_j is the one we suspect is causing these peculiarities we consider the sine transform of this wave separately. Its power spectrum is shown in Figure 6.6. As can be seen, the majority of the amplitude is at $k = 1$, with a cascade of amplitude through the higher wavenumbers. This cascade effect is due to the convergence properties of the sine transform, as discussed for the DFT in Section 6.3.2.

To understand Figure 6.5 better we consider again the definition of the sine transform. From equation (6.18) the sine transform is always zero at $k = 0$ so we only consider $k \geq 1$.

We first consider the sine transform at $k = 1$. Substituting $k = 1$ and $L = 1/4$ into equation (6.18) gives $\overline{c} = \overline{d} = 0$ and $\overline{b} = N$ from Lemma 3.1 but $\overline{a}$ does not fit the form of the orthogonality relation so has to be treated differently.

$$\overline{a} \;=\; -\frac{1}{2}\sum_{j=0}^{N-1}\left[e^{\frac{i\pi 3j}{2N}} - e^{\frac{-i\pi j}{2N}} - e^{\frac{i\pi j}{2N}} + e^{\frac{-i\pi 3j}{2N}}\right].$$

so for $N = 64$ we have

$$\overline{a} = -\frac{1}{2}\sum_{j=0}^{63}\left[e^{\frac{i\pi 3j}{128}} - e^{\frac{-i\pi j}{128}} - e^{\frac{i\pi j}{128}} + e^{\frac{-i\pi 3j}{128}}\right],$$

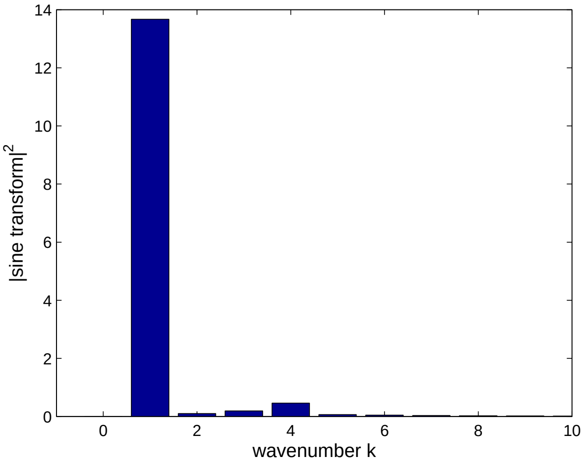


Figure 6.5: The power spectrum of the sine transform of f_j over the domain $L = 1/4$, scaled by a factor of $2/N$.

which, after applying the scaling factor $2/N$, corresponds to the amplitude we see at $k = 1$ in Figure 6.6(a) and

$$\widetilde{f}_1 = -\frac{1}{2}\sum_{j=0}^{63}\left[e^{\frac{i\pi 3j}{128}} - e^{\frac{-i\pi j}{128}} - e^{\frac{i\pi j}{128}} + e^{\frac{-i\pi 3j}{128}}\right] + 64,$$

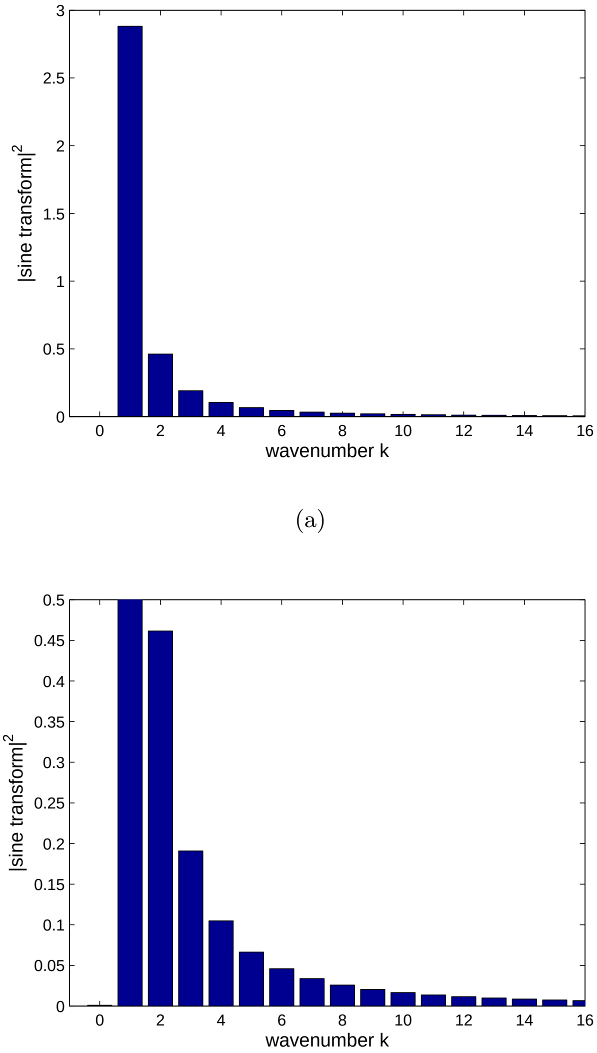
which, after applying the scaling factor $2/N$, corresponds to the amplitude we see at $k = 1$ in Figure 6.5.

The value of $\widetilde{f}_k$ at $k = 2, 3, 4, \cdots$ can be calculated in the same way and show that the apparent peculiarities in Figure 6.5 are caused by the long-wave.

As with the DFT, there is a cascading effect of the 'long-wave' information through the wavenumbers contained by the spectrum. In comparison to the DFT, instead of being sent to wavenumber $k = 0$, in the sine transform the majority of the long-wave information is sent to wavenumber $k = 1$.

6.5 Summary

We have discussed the effect the LAM domain has on the DFT and illustrated that the same wave on domains of different size will have a different wavenumber. We have demonstrated for a general case that, when using the DFT, the long-wave information is aliased onto other wavenumbers, with the majority being aliased onto wavenumber



(a)

(b)

Figure 6.6: (a) The power spectrum of the sine transform of $2\sin(2\pi x_j)$ over the domain $L = 1/4$, scaled by a factor of $2/N$. (b) Close up of $0 - 0.5$ amplitude section.

$k = 0$ and then cascading down through the spectrum. In comparison, when using the sine transform, the magnitude at $k = 0$ is automatically set to zero and the majority of the long-wave information is instead sent to wavenumber $k = 1$.

Chapter 7

A 1D linear advection-diffusion model

We wish to model a simple PDE in order to mimic the operational system in a simple model. We want our model to have as many of the properties of the operational model as possible in order to be able to understand the potential problems. However, by using a simple 1D-PDE model we are more able to identify the cause of problems and investigate them individually.

We have pre-existing code for a parent model of the 1D heat equation [4] and this is adapted here to include a nested LAM model. The heat equation is also extended to the 1D linear advection-diffusion equation.

7.1 The advection-diffusion equation

Mathematical models involving a combination of advection and diffusion are widespread. For example, in meteorology the advection-diffusion equation can be used to model the dispersal of atmospheric pollution [62, p.3].

We use the 1D linear advection-diffusion equation [62, p.12]

$$u_t + cu_x = \sigma u_{xx}, \qquad t \geq 0, \tag{7.1}$$

on the domain $x \in (0, 1]$, with periodic boundary conditions

$$u(x,t) = u(x+1,t), \tag{7.2}$$

where $u = u(x,t)$ is the temperature, x is the spatial coordinate, t is time, c is the advection velocity and $\sigma \geq 0$ is the diffusion constant.

7.2 The discrete problem

The continuous problem (7.1) is now approximated by a discrete problem. Equation (7.1) can be discretised using an explicit Euler discretisation for the time derivative, centered difference for the diffusion and upwind difference for the advection [23, p.138]. The resulting finite-difference approximation to equation (7.1) is

$$\frac{u_{j,n+1} - u_{j,n}}{\Delta t} = -c\left(\frac{u_{j,n} - u_{j-1,n}}{\Delta x}\right) + \sigma\left(\frac{u_{j+1,n} - 2u_{j,n} + u_{j-1,n}}{(\Delta x)^2}\right), \tag{7.3}$$

with boundary conditions

$$u_{0,n} = u_{N,n}, \tag{7.4}$$

where $u_{j,n} = u(x_j, t_n)$ and $x_j = j\Delta x$, $t_n = n\Delta t$ where Δx is the grid length and Δt is the timestep. It is worth noting here that since we discretise the advection term using an upstream scheme, we assume $c \geq 0$.

Equation (7.3) can be rewritten succinctly as

$$u_{j,n+1} = (\nu + \mu)u_{j-1,n} + (1 - \nu - 2\mu)u_{j,n} + \mu u_{j+1,n}, \tag{7.5}$$

where

$$\nu = \frac{c\Delta t}{\Delta x} \text{ and } \mu = \frac{\sigma\Delta t}{(\Delta x)^2}.$$

Before we use this finite difference scheme we need to know if it is accurate and stable for the problem we are considering.

7.2.1 Accuracy

To check the accuracy of the finite difference approximation (7.5) we calculate the truncation error of the scheme.

Assuming u is a solution of the continous problem the truncation error $\tau_{j,n}$ is

$$\begin{aligned}\tau_{j,n} &= \frac{u(j\Delta x,(n+1)\Delta t)-u(j\Delta x,n\Delta t)}{\Delta t}\\ &+c\left(\frac{u(j\Delta x,n\Delta t)-u((j-1)\Delta x,n\Delta t)}{\Delta x}\right)\\ &-\sigma\left(\frac{u((j+1)\Delta x,n\Delta t)-2u(j\Delta x,n\Delta t)+u((j-1)\Delta x,n\Delta t)}{(\Delta x)^2}\right), \quad (7.6)\end{aligned}$$

where $\{u(j\Delta x,n\Delta t)\}$ is the set of values obtained by restricting the exact solution $u(x,t)$ to the nodal points x_j,t_n [45, p.368], [48, p.296].

Using a Taylor expansion about $j\Delta x$, $n\Delta t$ gives

$$\begin{aligned}u(j\Delta x,(n+1)\Delta t) &= u(x_j,t_n+\Delta t)\\ &= u+(\Delta t)u_t+\frac{(\Delta t)^2}{2!}u_{tt}+O(\Delta t^3)\end{aligned}$$

$$\begin{aligned}u((j\pm 1)\Delta x,n\Delta t) &= u\pm(\Delta x)u_x+\frac{(\pm\Delta x)^2}{2!}u_{xx}+\frac{(\pm\Delta x)^3}{3!}u_{xxx}\\ &+\frac{(\pm\Delta x)^4}{4!}u_{xxxx}+O(\Delta x^5).\end{aligned}$$

To ease notation, in all the truncation error analysis we write u for $u(j\Delta x,n\Delta t)$, u_t for $u_t(j\Delta x,n\Delta t)$, u_x for $u_x(j\Delta x,n\Delta t)$ and so on.

Substituting the Taylor expansions into equation (7.6) gives, after cancellation,

$$\begin{aligned}\tau_{j,n} &= u_t+\frac{\Delta t}{2}u_{tt}+\frac{(\Delta t)^2}{3!}u_{ttt}+O(\Delta t^3)\\ &+c\left[u_x-\frac{(\Delta x)}{2}u_{xx}+\frac{(\Delta x)^2}{3!}u_{xxx}-\frac{(\Delta x)^3}{4!}u_{xxxx}+O(\Delta x^4)\right]\\ &-\sigma\left[u_{xx}+\frac{(\Delta x)^2}{12}u_{xxxx}+O(\Delta x^4)\right]. \qquad (7.7)\end{aligned}$$

Now from equation (7.1) we know $u_t+cu_x-\sigma u_{xx}=0$, which leaves

$$\begin{aligned}\tau_{j,n} &= \frac{\Delta t}{2}u_{tt}+\frac{(\Delta t)^2}{3!}u_{ttt}-c\frac{(\Delta x)}{2}u_{xx}+c\frac{(\Delta x)^2}{3!}u_{xxx}\\ &-\left[c\frac{(\Delta x)^3}{4!}+\sigma\frac{(\Delta x)^2}{12}\right]u_{xxxx}+O(\Delta t^3,\Delta x^4). \qquad (7.8)\end{aligned}$$

From equation (7.8) we can see that

$$\tau_{j,n}\to 0 \text{ as } \begin{cases}\Delta t & \to 0\\ \Delta x & \to 0\end{cases} \qquad (7.9)$$

and that $\tau_{j,n}$ is $O(\Delta x,\Delta t)$.

Considering the truncation error (7.7), we can see that equation (7.3) is also a second order accurate in space, first order accurate in time approximation to the modified equation

$$\frac{\partial u}{\partial t}+c\frac{\partial u}{\partial x}=\sigma\left(1+\frac{Pe}{2}\right)\frac{\partial^2 u}{\partial x^2}, \tag{7.10}$$

where $Pe = c\Delta x/\sigma$ is the grid Peclet number [23, p.138]. Comparing the modified equation (7.10) with the original advection-diffusion equation (7.1) shows that the discretisation scheme (7.3) generates an inaccurate approximation to the diffusion term unless $Pe/2$ is sufficiently small [23, p.138]. The Peclet number is a nondimensional parameter classically defined as the ratio of the strength of advection to diffusion. Since here the length scale is the grid spacing, Pe is a measure of the relative strengths of advection and diffusion at the smallest spatial scales resolved on the grid [23, p.138].

7.2.2 Stability

As well as being accurate we also require a stable scheme. To investigate the stability of the scheme we use a Fourier stability analysis [23, p.43-45]. Let

$$u_{j,n}=a_n e^{ikj\Delta x}. \tag{7.11}$$

Substituting this into equation (7.3) gives

$$\begin{aligned}a_{n+1}e^{ikj\Delta x}-a_n e^{ikj\Delta x} &= -\nu\left(a_n e^{ikj\Delta x}-a_n e^{ik(j-1)\Delta x}\right)\\ &\quad+\mu\left(a_n e^{ik(j+1)\Delta x}-2a_n e^{ikj\Delta x}+a_n e^{ik(j-1)\Delta x}\right),\end{aligned}$$

which after cancellation and rearrangement can be written as

$$a_{n+1}=a_n\left[1-\nu\left(1-e^{-ik\Delta x}\right)+\mu\left(2\cos\left(k\Delta x\right)-2\right)\right].$$

Replacing the exponential term with sines and cosines and then rearranging, this can be written as

$$a_{n+1}=a_n\left[1-\left(\nu+2\mu\right)\left(1-\cos\left(k\Delta x\right)\right)-i\nu\sin\left(k\Delta x\right)\right],$$

or more succinctly as

$$a_{n+1}=G(k)a_n,$$

where

$$G(k)\overset{def}{=}1-\left(\nu+2\mu\right)\left(1-\cos\left(k\Delta x\right)\right)-i\nu\sin\left(k\Delta x\right),$$

is known as the amplification factor. For a scheme to be absolutely stable we require $|G(k)| \leq 1$. Therefore, we have the stability condition

$$|1 - (\nu + 2\mu)(1 - \cos(k\Delta x)) - i\nu \sin(k\Delta x)| \leq 1.$$

If we note here that if $\sqrt{a} \leq 1$ then $a \leq 1$, then we need

$$[1 - (\nu + 2\mu)(1 - \cos(k\Delta x))]^2 + \nu^2 \sin^2(k\Delta x) \leq 1.$$

We can express everything in terms of cosines using the trigonometric formula $\sin^2 a + \cos^2 a = 1$, and if we also note that $\nu = \mu Pe$ then we can write the inequality as

$$[1 - (\mu Pe + 2\mu)(1 - \cos(k\Delta x))]^2 + (\mu Pe)^2 \left(1 - \cos^2(k\Delta x)\right) \leq 1.$$

Multiplying out the first term, factorizing the second, cancelling the 1 on both sides and pulling out the common factor $(1 - \cos(k\Delta x))$ gives

$$(1 - \cos(k\Delta x)) \Big[(\mu Pe + 2\mu)^2 (1 - \cos(k\Delta x)) - 2(\mu Pe + 2\mu) + (\mu Pe)^2 (1 + \cos(k\Delta x))\Big] \leq 0.$$

Now $1 - \cos(k\Delta x) \geq 0$ for all values of $k\Delta x$ so we need

$$(\mu Pe + 2\mu)^2 (1 - \cos(k\Delta x)) - 2(\mu Pe + 2\mu) + (\mu Pe)^2 (1 + \cos(k\Delta x)) \leq 0,$$

which after rearranging becomes

$$\left(4\mu^2 + 4\mu^2 Pe\right)(1 - \cos(k\Delta x)) + 2\mu^2 Pe^2 \leq 2(\mu Pe + 2\mu).$$

Here note that $1 - \cos(k\Delta x) \in [0, 2]$, so it is sufficient to have

$$2\left(4\mu^2 + 4\mu^2 Pe\right) + 2\mu^2 Pe^2 \leq 2(\mu Pe + 2\mu),$$

which can also be written as

$$(\mu Pe + 2\mu)^2 \leq \mu Pe + 2\mu.$$

Now since $(\mu Pe + 2\mu) \geq 0$ we can divide through and are left with

$$\mu(Pe + 2) \leq 1.$$

Now note that $\nu = \mu Pe$ and we have

$$\nu + 2\mu \leq 1,$$

or

$$\frac{\Delta t(2\sigma + c\Delta x)}{\Delta x^2} \leq 1, \tag{7.12}$$

as a sufficient condition for stability.

It is worth noting here that if we set $\sigma = 0$ in equation (7.12) then we get

$$\nu \leq 1,$$

the standard stability condition for the advection equation [23, p.45]. If we instead set $c = 0$ in equation (7.12) then we get

$$\mu \leq \frac{1}{2},$$

the standard stability condition for the heat equation [62, p.17].

Now that we know the accuracy and stabilty conditions of the scheme, we can apply the discrete equations to a parent and LAM domain.

7.3 The model design

We now use the discrete equation (7.5) to approximate equation (7.1) on the parent and LAM domains. On the parent domain we use u^P, x^P, t^P as the temperature, space and time coordinates respectively and on the LAM we use u^L, x^L, t^L.

7.3.1 The parent model

The parent domain $x^P \in (0, 1]$ is divided up to contain N spatial gridpoints and there are $\mathcal{T}$ parent timesteps. On the parent grid equation (7.5) becomes

$$u^P_{j,n+1} = (\nu^P + \mu^P)u^P_{j-1,n} + (1 - \nu^P - 2\mu^P)u^P_{j,n} + \mu^P u^P_{j+1,n}, \tag{7.13}$$

where

$$\nu^P = \frac{c\Delta t^P}{\Delta x^P} \text{ and } \mu^P = \frac{\sigma \Delta t^P}{(\Delta x^P)^2},$$

with boundary conditions

$$u^P_{0,n} = u^P_{N,n}. \tag{7.14}$$

Here $u^P_{j,n} = u^P(x^P_j, t^P_n)$, $x^P_j = j\Delta x^P$, and $t^P_n = n\Delta t^P$. The gridspacing $\Delta x^P = 1/N$ and the timestep $\Delta t^P = 0.5/\mathcal{T}$.

7.3.2 The LAM

The LAM domain is nested within the parent domain. The design of the LAM domain can be seen in Figure 5.1. The LAM covers D parent grid spaces where $D = B2 - B1$ and $B1, B2$ are the parent gridpoints corresponding to the boundaries of the limited area domain. Therefore, we have the boundary conditions

$$x_1^L = x_{B1}^P, \quad x_M^L = x_{B2}^P. \tag{7.15}$$

M is the number of spatial gridpoints in the LAM and S is the number of LAM timesteps. The spatial and temporal resolutions of the LAM are defined by their ratio to the parent resolutions. If Δx^P is the parent gridspacing then the LAM gridspacing is $\Delta x^L = \Delta x^P/h$, and if the parent timestep is Δt^P then the LAM timestep is $\Delta t^L = \Delta t^P/\tau$. Therefore $M = (D \times h) + 1$ and $S = \mathcal{T} \times \tau$. Once the value of h has been chosen, the value of τ is chosen such as to keep the value of μ^L the same as μ^P. This is done because the scheme is second order accurate in space[1] and first order accurate in time so we wish to keep the value of $\Delta t/\Delta x^2$ constant.

On the LAM grid equation (7.5) becomes

$$u_{j,n+1}^L = (\nu^L + \mu^L)u_{j-1,n}^L + (1 - \nu^L - 2\mu^L)u_{j,n}^L + \mu^L u_{j+1,n}^L, \tag{7.16}$$

where

$$\nu^L = \frac{c\Delta t^L}{\Delta x^L} \text{ and } \mu^L = \frac{\sigma \Delta t^L}{(\Delta x^L)^2}.$$

However, due to the design of the LAM domain we have the boundary conditions

$$u_{1,n}^L = u_{B1,n/\tau}^P, \quad u_{M,n}^L = u_{B2,n/\tau}^P. \tag{7.17}$$

When the LAM is run with a smaller timestep than that of the parent model, the parent model output is interpolated to the LAM timesteps at gridpoints $B1$ and $B2$ to provide the $u_{B1,n/\tau}^P$, $u_{B2,n/\tau}^P$ values.

In order to relax the solution on the interior of the LAM domain to the values prescribed at the boundaries there is a buffer zone implemented at the boundaries of the LAM, as described in Section 5.1.

[1] if Pe is sufficiently small, as discussed in Section 7.2

7.3.3 Convergence of the model

Now that we have a model we need to check that it converges and that it is converging to the truth. One advantage of the advection-diffusion equation is that is is possible to find an analytic solution. This makes testing convergence very simple. We consider the 1D advection-diffusion equation

$$u_t + cu_x = \sigma u_{xx}, \qquad t \geq 0, \tag{7.18}$$

on the domain $x \in (0, 1]$, subject to periodic boundary conditions

$$u(0, t) = u(1, t). \tag{7.19}$$

We try a solution of the form

$$u_\omega(x, t) = e^{i\omega x} T(t). \tag{7.20}$$

Note here that ω is chosen such as to satisfy the periodic boundary conditions, so it can take the value of any integer multiple of 2π

$$\omega = 2\pi n \qquad \text{where} \quad n = 0, \pm 1, \pm 2, \cdots .$$

Substituting this solution into equation (7.18) gives a first order ODE for T,

$$T' + T\left(ic\omega + \sigma\omega^2\right) = 0. \tag{7.21}$$

Substituting a solution of the form $e^{\lambda t}$ into equation (7.21) yields

$$\lambda = -\left(ic\omega + \sigma\omega^2\right)$$

and therefore

$$T(t) = e^{-\left(ic\omega + \sigma\omega^2\right)t}.$$

A solution is now

$$u_\omega(x, t) = e^{i\omega x} e^{-\left(ic\omega + \sigma\omega^2\right)t},$$

therefore the general solution can be written as a Fourier type solution

$$u(x, t) = \sum_{n=-\infty}^{\infty} a_n e^{i2\pi n(x - ct)} e^{-\sigma(2\pi n)^2 t}, \tag{7.22}$$

where a_n are constants to be determined by the initial conditions.

For example, if we have the initial conditions

$$u(x,0) = \sin(4\pi x), \tag{7.23}$$

then substituting into equation (7.22) gives

$$u(x,0) = \sin(4\pi x) = \sum_{n=-\infty}^{\infty} a_n e^{i2\pi n x}. \tag{7.24}$$

Using Fourier orthogonality relationships gives

$$a_n = \begin{cases} -i/2 & \text{when } n = 2 \\ i/2 & \text{when } n = -2 \\ 0 & \text{otherwise.} \end{cases} \tag{7.25}$$

Therefore

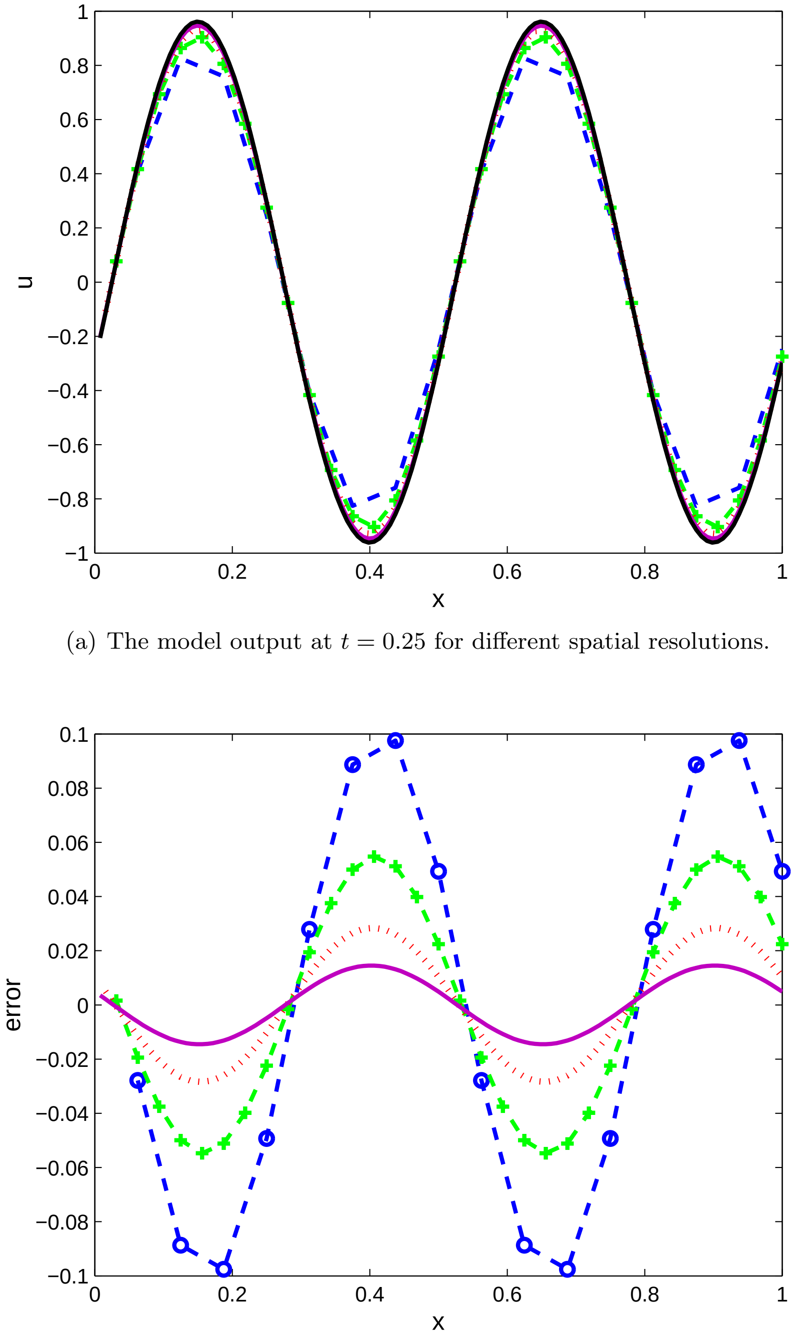
$$\begin{aligned} u(x,t) &= \frac{-i}{2}\left(e^{i4\pi(x-ct)} - e^{-i4\pi(x-ct)}\right) e^{-\sigma 16\pi^2 t}, \\ &= \sin(4\pi(x-ct))e^{-\sigma 16\pi^2 t}, \end{aligned} \tag{7.26}$$

in this example. By running the model from the initial conditions (7.23) we can now compare the model output with this analytic solution.

Figure 7.1(a) shows the analytic solution (7.26) (black line) and the model solutions when run at four different spatial resolutions, at time $t = 0.25$. We take $\sigma = 0.001$, $c = 0.1$ and we take Δt such as to keep the value of μ constant. The spatial resolutions are $\Delta x_1 = 1/16$ (blue), $\Delta x_2 = 1/32$ (green), $\Delta x_3 = 1/64$ (red) and $\Delta x_4 = 1/128$ (pink). As can be seen, as Δx decreases the model converges to the analytic solution. It is also worth noting is that the model is more diffusive at lower resolutions. The convergence of the model is further confirmed if we consider the errors. Figure 7.1(b) shows the error in the model outputs, where the error is model output minus analytic solution. As we see, as Δx decreases so to does the error. This example demonstrates that we do have a convergent model and that it is converging to the truth.

7.4 Observations

To run any data assimilation scheme we need observational data from somewhere. As we have no experimental data to use for observations, we run twin experiments; The



(a) The model output at $t = 0.25$ for different spatial resolutions.

(b) Error in model outputs compared to the analytic solution

Figure 7.1: In (a) the black line is the analytic solution, in (a) and (b) the dashed blue line with cirles is the $\Delta x = 1/16$ case, the dashed green line with crosses is the $\Delta x = 1/32$ case, the dotted red line is the $\Delta x = 1/64$ case and the solid pink line is the $\Delta x = 1/128$ case.

parent model is run once at a high resolution to generate a reference trajectory which is then treated as the truth. From this reference trajectory we take artifical observations which are then used in the data assimilation on both the parent and the LAM. The parent model is then run a second time at a lower resolution and from different initial conditions. It is this model run which is referred to as the parent model. The resolution of the reference trajectory (truth) is defined by its ratio to the resolution of the LAM. The spatial resolution is $\Delta x^{truth} = \Delta x^L/g$ and the temporal resolution is $\Delta t^{truth} = \Delta t^L/\rho$. The values of g and ρ are chosen to keep $\mu^{truth} = \mu^L$.

7.5 The data assimilation algorithm

Having considered the model itself we now implement our 4D-Var data assimilation algorithm on both the parent and LAM models. The original parent model [4] of the 1D heat equation already has a working gridpoint (physical) 4D-Var scheme so we adapt this code to work for our advection-diffusion parent model. We also adapt it to both an incremental gridpoint (Section 5.4.1) and a spectral (Section 5.4.2) scheme for the LAM.

As discussed in Section 2.3.1, to generate an analysis using 4D-Var we must minimise the cost function

$$J(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x^b})^T \boldsymbol{B^{-1}}(\boldsymbol{x} - \boldsymbol{x^b}) + \frac{1}{2}\sum_{k=0}^{T}(\boldsymbol{y_k} - \boldsymbol{H}\boldsymbol{x_k})^T \boldsymbol{R_k^{-1}}(\boldsymbol{y_k} - \boldsymbol{H}\boldsymbol{x_k}),$$

subject to the model equation

$$\boldsymbol{x_k} = \boldsymbol{M}\boldsymbol{x_{k-1}}.$$

To do this we need a minimisation algorithm. The original code [4] uses the CONMIN program [76], which allows the user the choice of using either a conjugate gradient method [73], [33, Sections 9.3, 10.2] or a Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [31], [77]. Throughout the work presented in this thesis we use the conjugate gradient option.

Both methods are iterative schemes and require the value of J and its gradient ∇J to be calculated at each iteration.

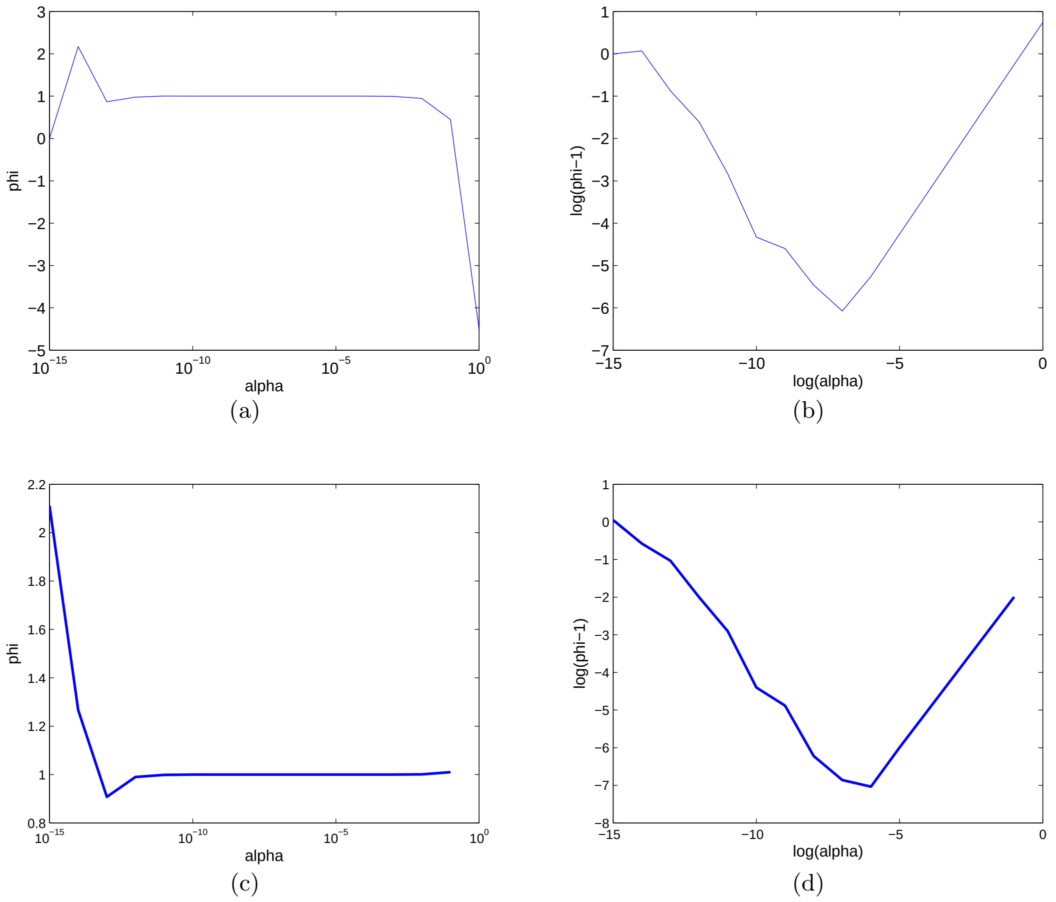
7.6 The gradient test

With an adjoint model coded for both the LAM and the parent model, we need to check that this adoint model is coded correctly. To do this we use the gradient test, as previously described in Section 2.3.2. The gradient test is run on both the parent model and the LAM, and on both the gridpoint and spectral data assimilation schemes.

Figure 7.2 shows an example result for the gridpoint scheme. The parent model is run with $N = 16$ gridpoints and 10 timesteps and the vector $\boldsymbol{\delta x}$ is generated using random data with unit norm. The LAM results shown are with the LAM covering parent gridpoints 8 to 16 and having four times the spatial resolution and sixteen times the temporal resolution of the parent model. The buffer zone width is 4. The LAM data assimilation is the incremental version. As can be clearly seen in Figure 7.2, for values of α between 10^{-3} and 10^{-13} we obtain a value of $\phi(\alpha)$ that is close to 1. This confirms that the cost function and the gradient of the cost function have been coded correctly in our gridpoint data assimilation scheme.

We now need to test the spectral scheme. Figure 7.3 shows an example result for the spectral scheme. The vector $\boldsymbol{\delta x}$ is again generated using random data with unit norm. Again the results shown are with the LAM covering half the domain of the parent model (gridpoints 8 to 16) with four times the spatial resolution and with 160 timesteps. The buffer zone width is 4. The matrix $\boldsymbol{\Sigma^{-1}}$ was set to equal the identity matrix. As can be clearly seen in Figure 7.2, for values of α between 10^{-3} and 10^{-13} we obtain a value of $\phi(\alpha)$ that is close to 1. This confirms that the cost function and the gradient of the cost function have been coded correctly in our spectral data assimilation scheme.

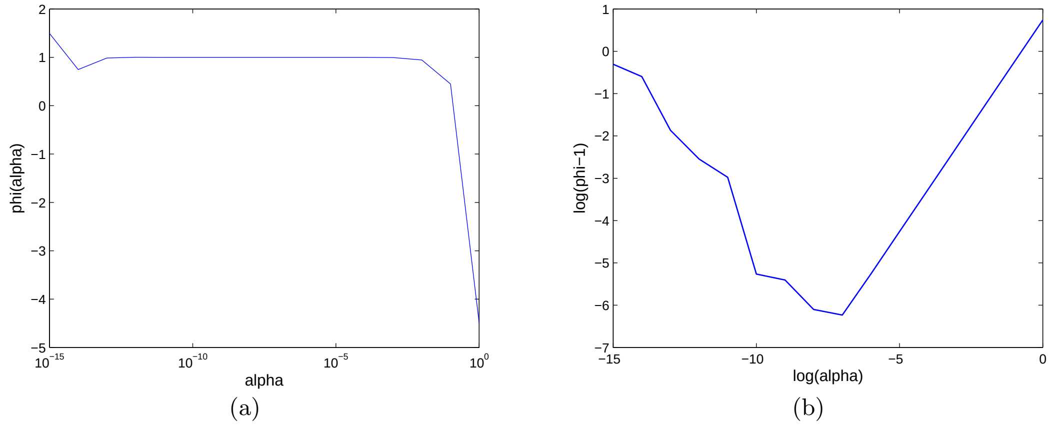
The test was run, on both schemes, for various different LAM domain sizes, for different ratios of space and time stepsize and different buffer zone widths. The unit value of $\phi(\alpha)$ over a range of α was observed in all cases.



(a) (b)

(c) (d)

Figure 7.2: Verification of the gradient calculation for the gridpoint scheme: (a) variation of $\phi(\alpha)$ with respect to (α) for the parent model, (b) variation of $\log(\phi - 1)$ with respect to $\log(\alpha)$ for the parent model, (c) variation of $\phi(\alpha)$ with respect to (α) for the LAM, (d) variation of $\log(\phi - 1)$ with respect to $\log(\alpha)$ for the LAM.



(a) (b)

Figure 7.3: Verification of the gradient calculation for the spectral scheme: (a) variation of $\phi(\alpha)$ with respect to (α) for the LAM, (b) variation of $\log(\phi - 1)$ with respect to $\log(\alpha)$ for the LAM.

7.7 Summary

We have introduced the 1D linear advection-diffusion model. We have demonstrated the stabilty and accuracy of the discretisation scheme and have also tested the accuracy of the data assimilation code using established methods. We now go on to use this model and the 4D-Var data assimilation scheme to investigate the behaviour of 4D-Var on a LAM grid.

Chapter 8

The representation of different scales in a 4D-Var analysis on a LAM domain

We aim to understand how different wavelengths are reproduced in a 4D-Var analysis on a LAM domain. We also investigate how the representation of a wave on the LAM domain compares to the same wave on the parent domain. To investigate these questions we consider a reference trajectory with a range of scales. It is this reference trajectory that is taken to be the truth, as described in Section 7.4, and it is this truth that we aim to recapture with our data assimilation.

Within a 4D-Var scheme the initial conditions (first guess) and the background $\boldsymbol{x^b}$ do not necessarily have to be the same [12]. Although throughout this Chapter the initial conditions are set equal to the background, in both the parent and LAM schemes, for clarity we will still distinguish between them.

As well as the background $\boldsymbol{x^b}$, valid at time $t = 0$, we also have the model trajectory generated by running the model forward from the background. This trajectory will be called the background trajectory.

Throughout this Chapter, the background for the LAM and the lateral boundary conditions (LBCs) are provided by the same parent model run, i.e. the LBCs are provided by the background trajectory. The background trajectory also provides the parent values used in the Davies relaxation (Section 5.1). The background trajectory may simply be from a run of the parent model from specified initial conditions or it may be from a parent analysis.

For all the results shown in this Chapter, the inverse background error covariance matrix $\boldsymbol{B}^{-1}$ is set equal to zero and therefore there is no background term in the cost function (i.e. $J_b = 0$). However, the background trajectory will still influence the LAM analysis through the boundary conditions and the buffer zone.

As described in Section 7.4, the observations are taken from a reference trajectory. In this chapter we take perfect observations with no added random noise or error. In both the parent and the LAM assimilation, we take an observation at every gridpoint and every timestep, unless otherwise stated. It is noted here that this means that the same observations are seen by the parent and LAM data assimilation, but the LAM also has some extra higher resolution observations not seen by the parent assimilation. Having observations at every gridpoint and every timestep will be referred to in this work as having ‘full observations’.

Although experiments have been run with a variety of domain values, in all the results to be presented here, we take our parent model to have 15 internal gridpoints and 10 timesteps in the assimilation window $t \in [0, 0.5]$. The LAM covers the right half of the parent domain, starting at the middle parent gridpoint. The spatial resolution of the LAM is four times that of the parent model and there are 16 LAM timesteps for every parent timestep. The buffer zone covers 4 LAM gridpoints. The spatial resolution of the reference trajectory, which provides our ‘truth’ as well as the observations, is twice that of the LAM and there are 4 timesteps for every LAM timestep. The constant of diffusivity is $\sigma = 0.001$ and the advection speed is $c = 0.1$. The observation error covariance matrix is taken to be $\boldsymbol{R} = \frac{N}{2}\boldsymbol{I}$ in the parent assimilation and $\boldsymbol{R} = \frac{M}{2}\boldsymbol{I}$ in the LAM assimilation, where N is the number of parent gridpoints and M is the number of LAM gridpoints, as described in Section 7.3. However, this choice of matrix $\boldsymbol{R}$ is arbitrary since we have perfect observations and no background term in the cost function.

Once we have generated a parent and a LAM analysis we compare both of these to the truth. We show how the parent and LAM compare to the truth by plotting the temperature $\boldsymbol{u}$ against the spatial coordinate $\boldsymbol{x}$. As well as comparing them in physical space we also take the DFT over the LAM domain, as described in Chapter 6, to compare the results in spectral space. We display the DFT coefficients in a power spectrum, plotted against wavenumber k. We compare the truth with the analyses generated by the data assimilation both at the initial time $t = 0$ and at the middle of the assimilation window, as this is when a 4D-Var analysis is most accurate [67].

We start by considering how different scales are treated in a 4D-Var assimilation using the gridpoint scheme (Section 5.4.1) and then go on to compare the results with those generated using the spectral scheme (Section 5.4.2).

Until otherwise stated we use the gridpoint assimilation scheme as defined in Section 5.4.1. The background and the initial conditions for the LAM assimilation are provided by the parent analysis, interpolated to the LAM grid. The parent analysis also provides the LBCs for the LAM.

8.1 Short waves with the gridpoint scheme

Although in this thesis we aim to understand how all the different scales are treated by a 4D-Var scheme, it is important to remember why we want to use high resolution models: To improve convective-scale forecasting. This requires a high resolution model which can resolve small-scale detail. However, if the LAM data assimilation cannot capture scales which are too short to be resolved by the parent, then we are achieving no benefit from using the higher resolution and are creating further problems by having to use a limited area domain.

We therefore begin by considering the treatment of waves which are too short to be resolved by the parent model and investigate whether the treatment of these waves is improved in the LAM data assimilation compared to that of the parent.

We have the reference trajectory

$$u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r),$$

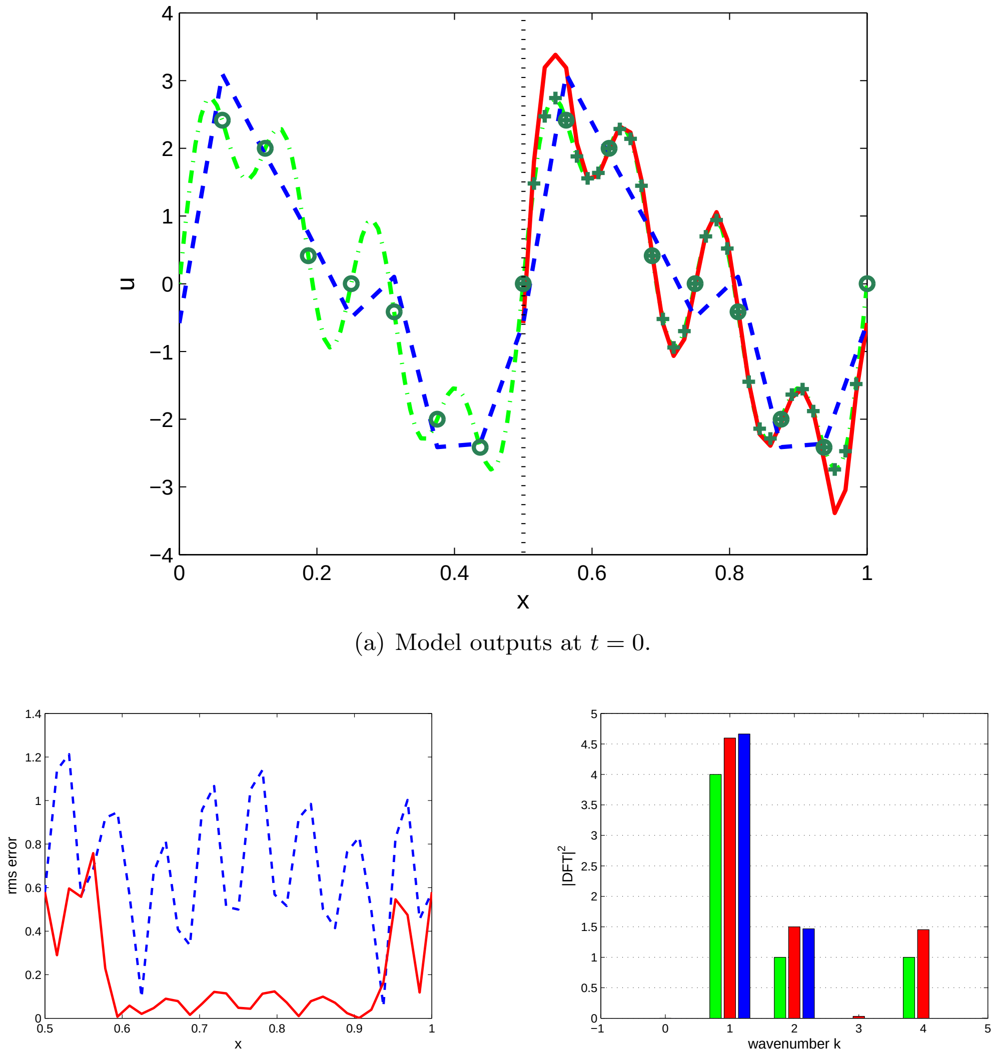
at time $t = 0$. This contains only waves whose wavelengths are shorter than the LAM domain, with the shortest wave being too short to be resolved by the parent model. We would expect this wave to be treated more accurately by the LAM data assimilation.

The initial condition for the parent scheme is

$$u^P(x_i^P, 0) = 2\sin(2\pi x_i^P).$$

Figure 8.1(a) shows the truth and both analyses in physical space, at the initial time $t = 0$. As can be seen, the higher resolution of the LAM compared with the parent allows it to pick up the higher resolution features in the truth that cannot be captured by the parent assimilation. The LAM analysis is an extremely good match to the truth everywhere except close to the boundaries. The LAM is able to fit to the oscillations in the truth caused by the small-scale wave. In contrast, the parent analysis is smoother and is unable to capture the small-scale detail, rendering it less accurate than the LAM analysis. This impression of increased accuracy in the LAM analysis is even more clearly demonstrated in Figure 8.1(b), which shows the rms (root mean square) error of both the LAM and parent analyses. To get a meaningful comparison we interpolate the parent analysis to the LAM grid and use this to calculate the error in the parent analysis. As can be seen, the LAM is significantly more accurate throughout the domain except for a couple of isolated points. It is also worth noting that the largest rms errors in the LAM analysis are close to the boundaries, where the LAM is being influenced by the parent through the boundaries and buffer zone. If we also calculate the average rms error over the LAM domain, we see that the LAM has an average rms error of is 0.2875, compared to the 0.7568 of the parent. This clearly demonstrates the improvement in accuracy we would expect due to the higher resolution of the LAM.

The benefits of the LAM's higher resolution can also be seen by considering the power spectrum, as shown in Figure 8.1(c). As we would expect from Section 6.3, the truth has amplitude in the power spectrum at wavenumbers $k = 1, 2, 4$. The parent analysis has amplitude at wavenumbers $k = 1, 2$ but has no amplitude at $k = 4$. This is because the parent grid does not have high enough resolution to capture this wave. In contrast, the higher resolution of the LAM grid has allowed the LAM analysis to capture all three waves, demonstrated by amplitude in the power spectrum at $k = 1, 2, 4$. This clearly illustrates that that the increased resolution of the LAM has allowed it to better fit the truth and produce a more accurate analysis. However, it is also interesting to note here

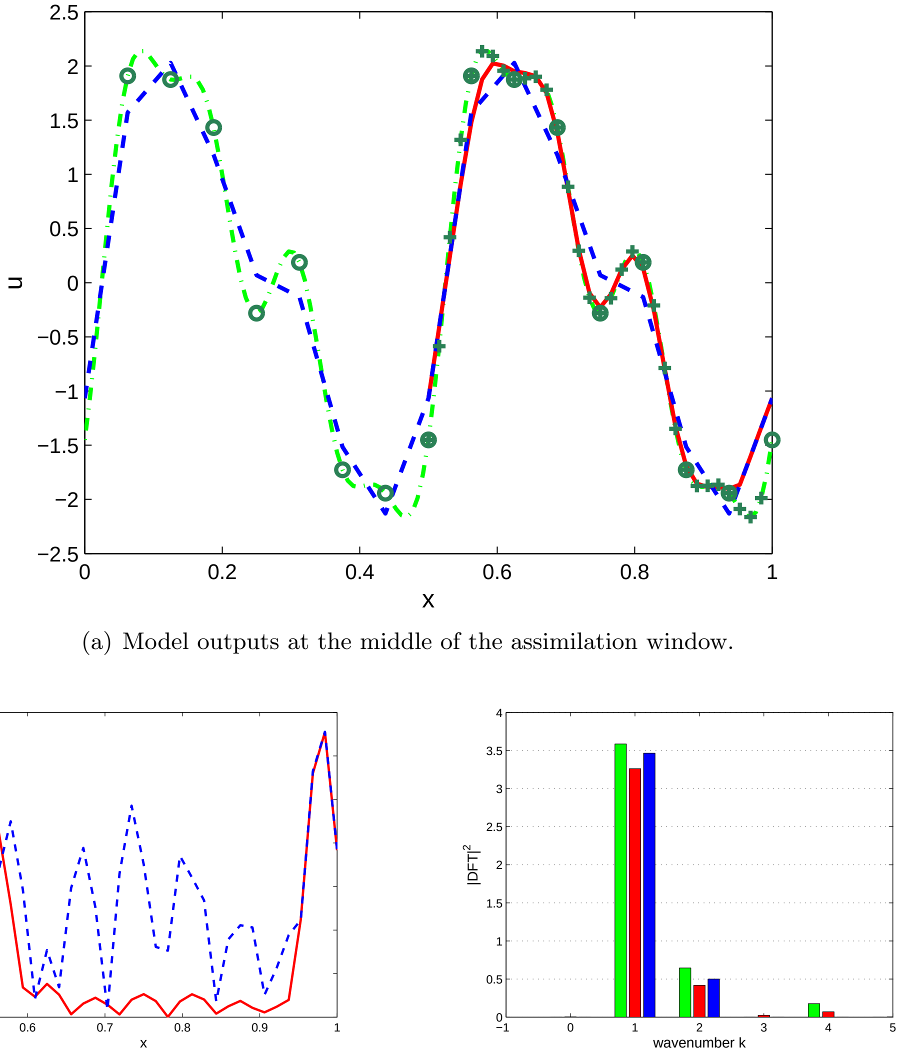


(a) Model outputs at $t = 0$.

(b) rms error between the truth and the model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

Figure 8.1: The truth is $u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. The vertical dotted line is the boundary of the LAM domain. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.



(a) Model outputs at the middle of the assimilation window.

(b) rms error between the truth and the model outputs at the middle of the assimilation window.

(c) Power spectrum at the middle of the assimilation window.

Figure 8.2: The truth at $t = 0$ is $u^r(x^r_j) = 2\sin(2\pi 2x^r_j) + \sin(2\pi 4x^r_j) + \sin(2\pi 8x^r_j)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.

that while the LAM is considered more accurate because it captures the $k = 4$ wave that is missed by the parent, for the $k = 1$ wave the LAM is only marginally better than the parent and for the $k = 2$ wave it is actually marginally worse.

If we now consider the middle of the assimilation window, we see in Figure 8.2(a) that the LAM analysis still looks better than the parent, as it fits to the truth more closely. The LAM matches the truth everywhere except close to the boundaries. This inaccuracy at the boundaries is due to the influence of the background trajectory at the LBCs and buffer zone. The higher accuracy of the LAM analysis compared to the parent analysis, is also confirmed by the rms error plotted in Figure 8.2(b).

If we consider the power spectrum, shown in Figure 8.2(c) we see that while the LAM is more accurate in the sense that it is capturing the $k = 4$ wave that is completely unresolved in the parent analysis, the LAM is actually less accurate than the parent in its representation of both the $k = 1$ and $k = 2$ waves. This is caused by the conflict in the LAM analysis of trying to fit to high resolution observations whilst also being constrained by inaccurate boundary conditions provided by the background trajectory.

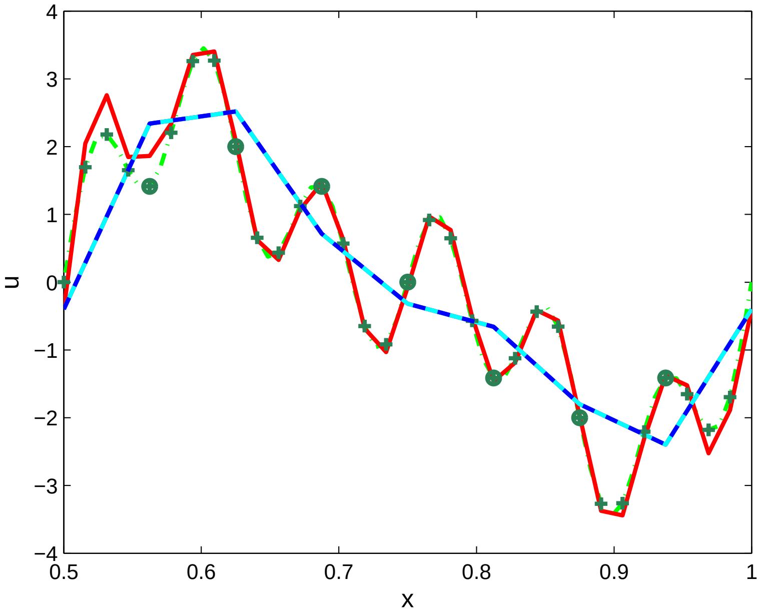
Figures 8.1 and 8.2 demonstrate the ability of the LAM to resolve finer scale detail than can be captured by the parent model. This is what we would hope and confirms the benefit of using high resolution models. However, there are further points of interest raised by Figures 8.1 and 8.2. These are now considered.

8.1.1 The benefit of data assimilation and high resolution observations

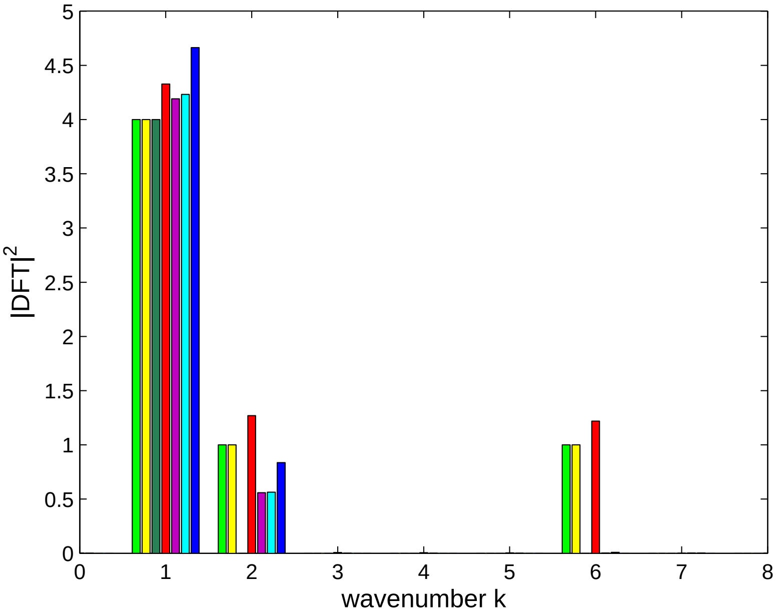
As we see in Figures 8.1 and 8.2, the higher resolution of the LAM allows it to resolve the small scale detail. However, it is worth noting the role of the data assimilation.

Although the resolution of the LAM makes resolving the small scales possible, simply running the LAM at this high resolution is not sufficient to achieve the high resolution detail: the data assimilation is also needed. This is illustrated in Figures 8.3(a) and 8.4(a), where the reference trajectory (truth) at $t = 0$ is

$$u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 12x_j^r).$$

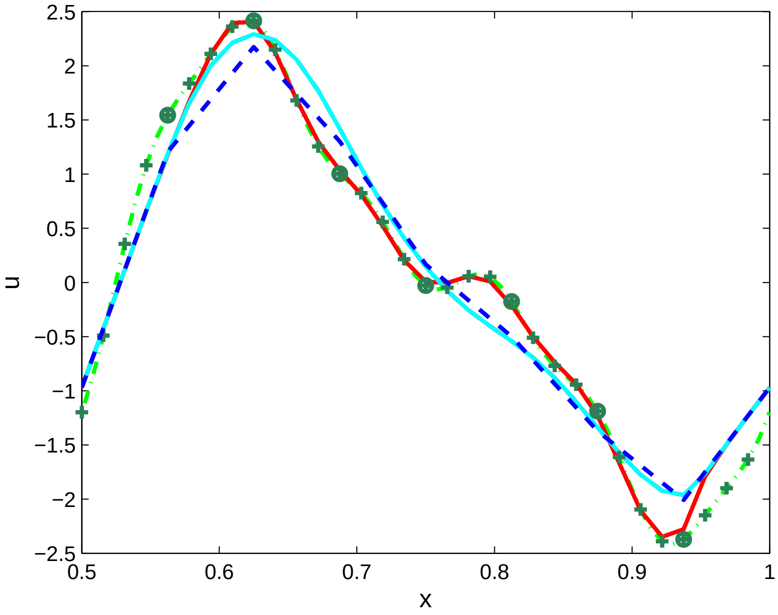


(a) Model outputs at $t = 0$.

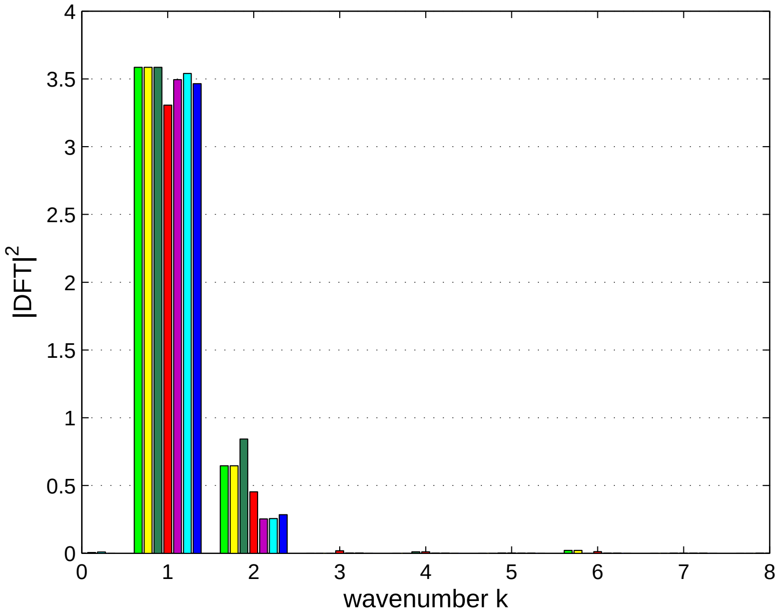


(b) Power spectrum at $t = 0$.

Figure 8.3: The truth at $t = 0$ is $u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 12x_j^r)$. In (a) Green is the truth, red is the LAM analysis run with full observations, pink is the LAM analysis run with low resolution observations, light blue is the LAM model run without data assimilation and dark blue is the parent analysis. The green crosses are the high resolution (full) LAM obs and the green circles are the parent (low) resolution observations. In (b) Green is the truth, yellow is the high resolution (full) LAM obs, dark green is the parent (low) resolution observations, red is the LAM analysis run with full observations, pink is the LAM analysis run with low resolution observations, light blue is the LAM model run without data assimilation and dark blue is the parent analysis.



(a) Model outputs at the middle of the assimilation window.



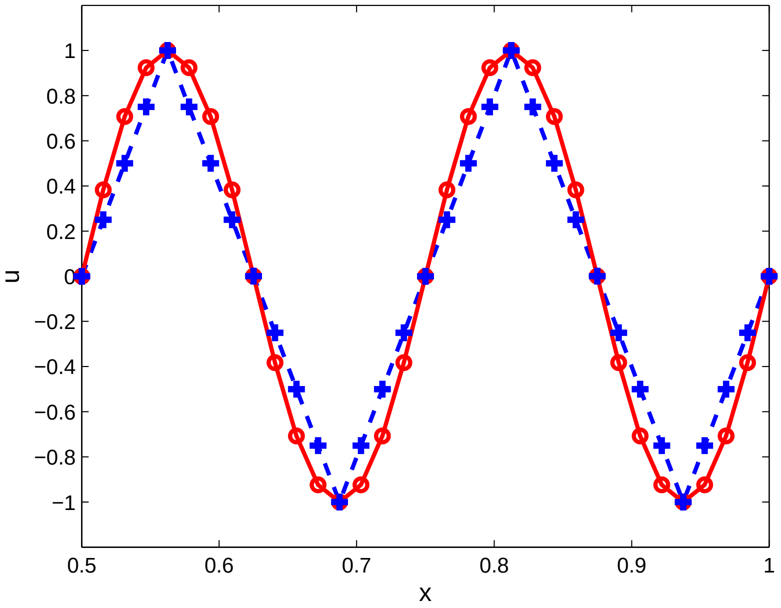
(b) Power spectrum at the middle of the assimilation window.

Figure 8.4: The truth at $t = 0$ is $u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 12x_j^r)$. In (a) Green is the truth, red is the LAM analysis run with full observations, pink is the LAM analysis run with low resolution observations, light blue is the LAM model run without data assimilation and dark blue is the parent analysis. The green crosses are the high resolution (full) LAM obs and the green circles are the parent (low) resolution observations. In (b) Green is the truth, yellow is the high resolution (full) LAM obs, dark green is the parent (low) resolution observations, red is the LAM analysis run with full observations, pink is the LAM analysis run with low resolution observations, light blue is the LAM model run without data assimilation and dark blue is the parent analysis.

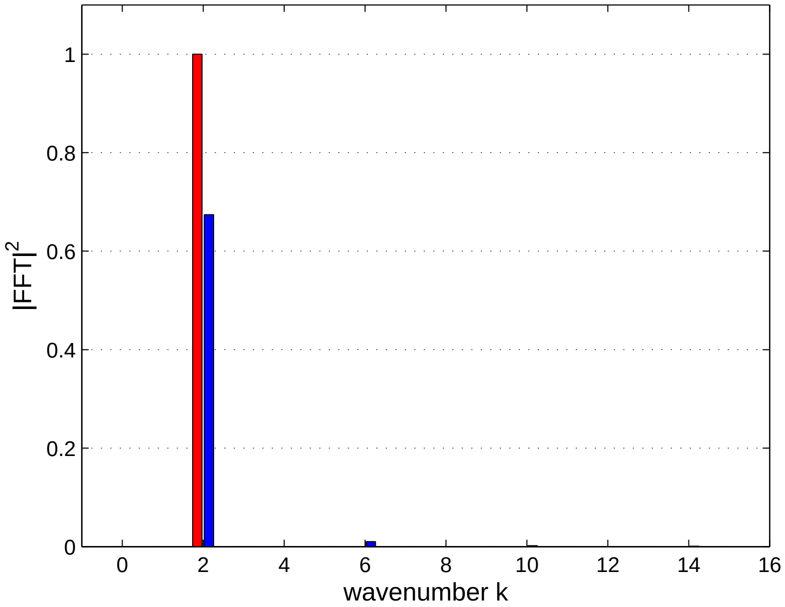
Plotted is the truth (green), the parent analysis (dark blue), the LAM analysis generated using full observations (red) and the LAM model run without data assimilation (light blue). In order to consider the benefit of high resolution observations, the LAM analysis generated with just the observations seen by the parent (i.e. low resolution observations) is also plotted (pink).

Considering $t = 0$ first, we begin by noting that the LAM model run without data assimilation matches the parent analysis exactly in physical space. This is as we would expect due to the parent analysis providing the initial conditions for the LAM. However, if we consider the power spectrum over the LAM domain, shown in Figure 8.3(b), we see that the power spectrum of these two outputs does not match in spectral space. This difference is caused by the resolution of the outputs. Although they appear the same in physical space, the LAM model is run from the parent analysis interpolated to the LAM grid. This change in resolution is what causes the differences in the power spectrum. It can be understood as follows: If we have the wave $\sin(2\pi 4x)$ on the parent grid then we get an amplitude of 1 at $k = 2$ in the power spectrum. Likewise, if we have the $\sin(2\pi 4x)$ wave on the LAM grid then we again get an amplitude of 1 at $k = 2$ in the power spectrum. However, if we interpolate the wave on the parent grid to the LAM resolution, we do not get an amplitude of 1 at $k = 2$ in the power spectrum, as demonstrated in Figure 8.5. This is because, as shown in Figure 8.5(a), in relation to the LAM resolution the interpolated wave is not a perfectly resolved $\sin(2\pi 4x)$ wave, it contains some error, and this error shows up in the power spectrum. This consequence of interpolating from the parent resolution to the LAM resolution and the resulting behaviour in the power spectrum is worth bearing in mind when we consider other results.

The implication for the LBCs is also worth noting. Even if the parent trajectory providing the LBCs is perfect, as is assumed with zero boundary conditions, it is perfect with respect to the parent resolution. As Figure 8.5(a) demonstrates, the corresponding interpolation is not perfect with respect to the LAM resolution. This error introduced by the differences in resolution is an issue for differences in the size of timesteps as well as the spatial steps. Therefore, errors will be introduced into the LAM through the LBCs.



(a) Difference between sine waves when generated at a higher resolution compared to simply interpolating a lower resolution.



(b) Differences in power spectra.

Figure 8.5: Difference between the sine wave $u(x_j) = \sin(2\pi 4 x_j)$ generated at a higher resolution (red) compared to simply interpolating a lower resolution wave (blue). (a) shows the difference in physical space and (b) shows the differnce in the power spectra.

We also notice from Figure 8.3(b) that the LAM analysis generated with low resolution observations has failed to capture the high resolution detail and remains equal to the parent analysis. This is due to the information content of the low resolution observations. On the parent grid, the high resolution detail in the observations is aliased to other wavenumbers; in the case of the $\sin(2\pi 12 x_j^r)$ wave this information is aliased to wavenumber $k = 2$. When we have the same low resolution observations on the LAM grid, although there is now the resolution to resolve the small scale wave, without extra high resolution information the small scale wave is still aliased.

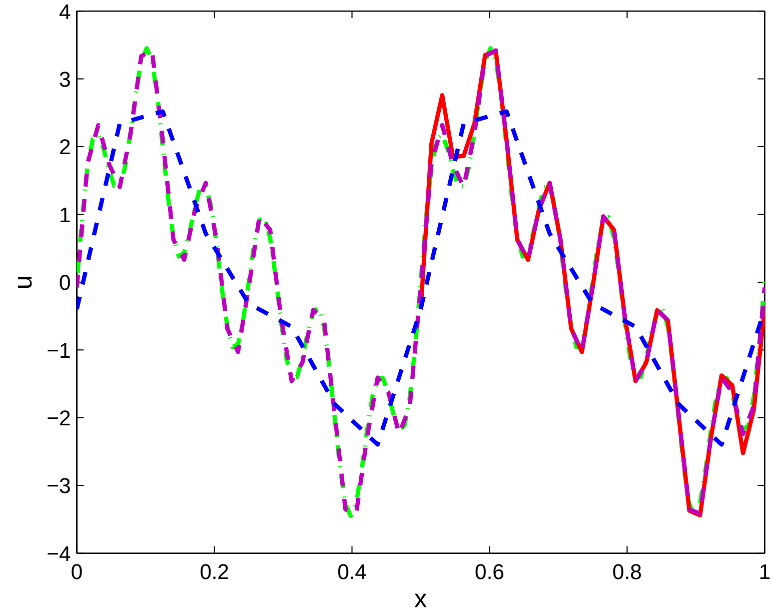
If we now consider the middle of the assimilation window, we see in Figure 8.4(a) that the LAM model run without data assimilation no longer matches the parent analysis, due to being run at a different resolution. However, the model run without data assimilation does match the LAM assimilation run with low resolution observations. This clearly demonstrates the need for high resolution observations. The only model trajectory that closely matches the truth is the LAM analysis run with full observations.

We also note that by the middle of the assimilation window there is no longer any amplitude in the power spectrum at $k = 6$, as can be seen in Figure 8.4(b). This is because the small scales have been damped by the model diffusion.

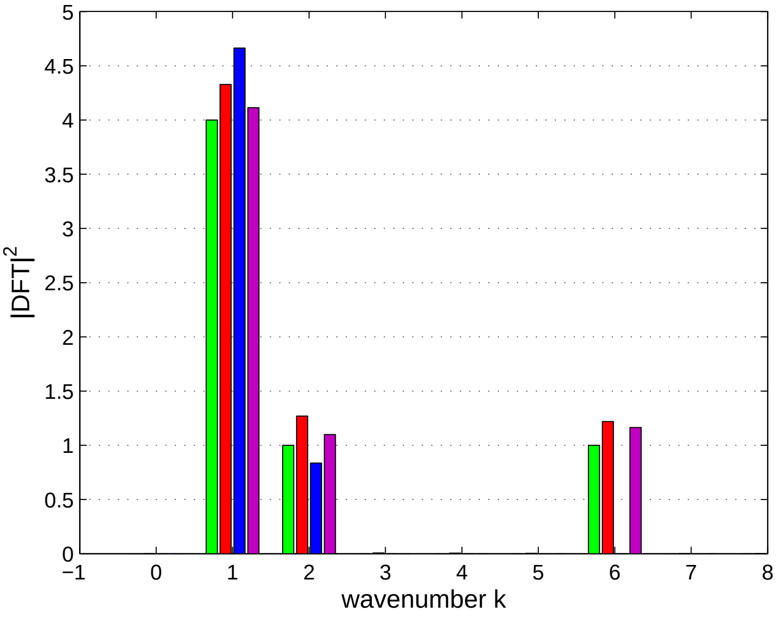
Figures 8.3 and 8.4 demonstrate that a high resolution model alone is not sufficient to improve the accuracy of the small scales. Data assimilation with high resolution observations is also needed if we are to represent the small scales accurately.

8.1.2 The ability of the LAM to capture the truth

Another important observation to be made from Figure 8.1 is that, although the LAM has the necessary resolution to resolve all the wavenumbers in the reference trajectory, we have perfect observations and there is no background term in the cost function, the LAM analysis does not fit the observations exactly. This is due to the influence of the background trajectory through the LBCs and buffer zone. In Figure 8.1(a) the LAM analysis matches the truth everywhere except close to the boundaries. The rms error plotted in Figure 8.1(b) also shows the largest errors in the LAM are close to the boundaries. These inaccuracies cause errors in the power spectrum of the LAM analysis.

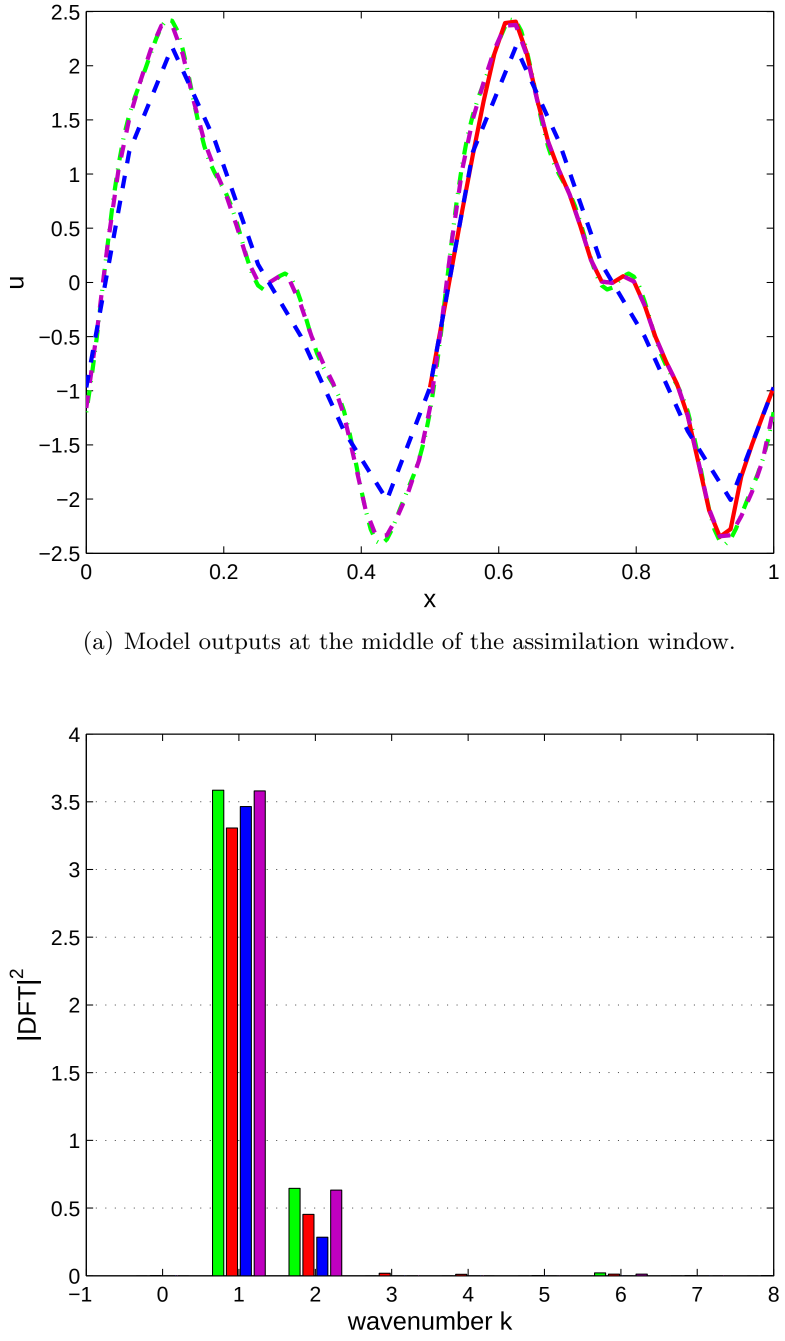


(a) Model outputs at $t = 0$.



(b) Power spectrum at $t = 0$.

Figure 8.6: Comparing the analysis generated at the same resolution on the parent and LAM domain. The truth at $t = 0$ is $u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 12x_j^r)$. Green is the truth, red is the LAM analysis on the LAM domain, pink is the analysis generated on the parent domain at high (LAM) resolution and blue is the low resolution parent analysis.



(a) Model outputs at the middle of the assimilation window.

(b) Power spectrum at the middle of the assimilation window.

Figure 8.7: Comparing the analysis generated at the same resolution on the parent and LAM domain. The truth at $t = 0$ is $u^r(x^r_j) = 2\sin(2\pi 2x^r_j) + \sin(2\pi 4x^r_j) + \sin(2\pi 12x^r_j)$. Green is the truth, red is the LAM analysis on the LAM domain, pink is the analysis generated on the parent domain at high (LAM) resolution and blue is the low resolution parent analysis.

Errors coming from the background trajectory mean the LAM cannot capture the truth.

To understand further the ability of the LAM to capture the truth, as well as generating a LAM analysis we now also generate a parent analysis at the same high resolution as the LAM, in order to provide a comparison without the influence of boundary conditions.

We again use the reference trajectory

$$u^r(x_j^r) = 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 12x_j^r),$$

at $t = 0$. The model outputs in physical space are shown in Figure 8.6(a). As we see, the parent analysis generated at the high (LAM) resolution can capture the truth accurately over the whole domain. In comparison, the LAM analysis has errors near the boundaries caused by the LBCs. If we also consider the power spectrum, as shown in Figure 8.6(b), the improved accuracy on the full domain is evident, particularly at low wavenumbers.

However, it should be noted that the high resolution parent analysis is still not an exact match to the truth. This is because of the model itself. At higher resolutions the model is less diffusive, as discussed in Section 7.3.3. The aim of 4D-Var is to find an analysis that, as well as being accurate at the initial time, also produces an accurate trajectory through the assimilation window. Due to this feature, the data assimilation is compensating for the discrepencies between the model runs. It over estimates the amplitudes at time $t = 0$ to account for the extra diffusion through the assimilation window at the LAM resolution. If we consider the power spectrum at the middle of the assimilation window, as shown in Figure 8.7(b), we see that the high resolution parent assimilation now does match the truth. In comparison, the LAM analysis does not, due to the influence of the LBCs.

Figure 8.7 also demonstrates that inaccuracies at different scales have different causes. At wavenumber $k = 6$, the error in the high resolution parent analysis is approximately the same as that for the LAM analysis. This is because, in both cases the error is caused by the data assimilation compensating for the higher diffusion rate at the LAM resolution. At $k = 1$ and $k = 2$ however, although the diffusion rate does cause the high resolution parent analysis to have some inaccuracies, those in the LAM analysis are of greater magnitude. This is because the LAM analysis also has errors at low wavenumbers due to the LBCs.

The errors introduced because of the LBCs are an important point to note, as it affects what we can expect to achieve with a LAM assimilation: even with a high enough resolution to resolve all the scales in the reference trajectory, the LAM cannot capture the truth, due to the errors introduced by the background trajectory at the boundaries and buffer zone.

8.1.3 Summary of the small scales

We have demonstrated that, when given high resolution observations, the higher resolution of the LAM does allow the data assimilation to capture small scale detail that cannot be resolved by the parent analysis.

However, we also demonstrated that the LAM analysis cannot capture the truth exactly, even with perfect observations and no background term, due to the errors from the LBCs.

We showed that the LAM analysis has errors on the low wavenumbers caused by having to constrain the analysis to match the LBCs. Due to the model being less diffusive at higher resolution the LAM analysis also has errors introduced by trying to compensate for the difference in diffusion rate compared to the truth. These errors occur on all wavenumbers but the high wavenumbers are affected to a lesser degree due to these scales diffusing away more rapidly.

Having considered the scales which are contained within the spectrum of the LAM, we now consider the treatment of long-waves, which have lengthscales longer than the domain of the LAM.

8.2 Long-waves with the gridpoint scheme

In Section 4.2.3 we introduced the long-wave problem. This is the problem of having waves whose wavelengths are longer than the domain of the model. These waves are referred to here as long-waves. We now consider how these long-waves are treated on a LAM domain.

We start by considering a case with a single long-wave.

8.2.1 One long-wave

We have the reference trajectory

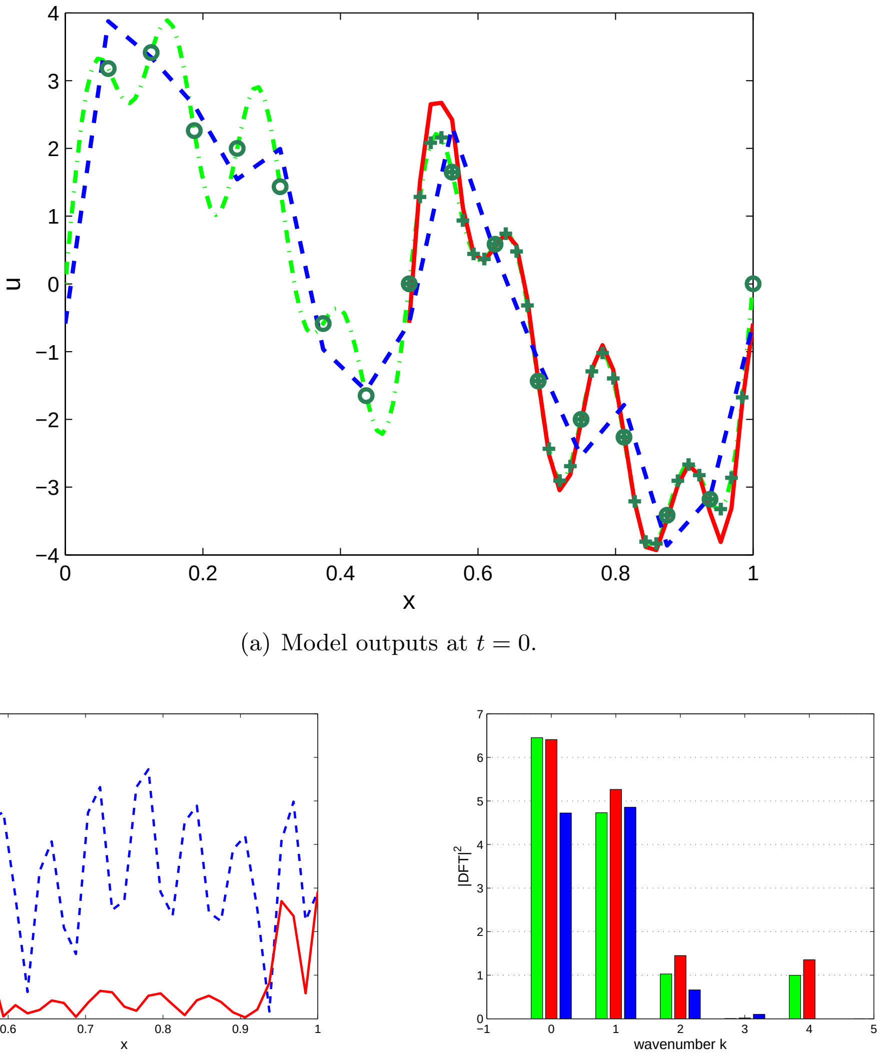
$$u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r),$$

at time $t = 0$. This contains one wave whose wavelength is longer than the domain of the LAM. The initial condition for the parent scheme is

$$u^P(x_i^P, 0) = 2\sin(2\pi x_i^P).$$

Figure 8.8(a) shows the truth and both analyses in physical space, at the initial time $t = 0$. As was the case in Section 8.1, the LAM analysis looks more accurate as it fits more closely to the truth. Again however, the LAM analysis has errors close to the boundaries due to the influence of the background trajectory. This is also confirmed by the plot of rms error shown in Figure 8.8(b) and by the average rms error values over the LAM domain, which are 0.2864 for the LAM analysis compared to the 0.7561 of the parent.

However, the truth contains a wave which is longer than the domain of the LAM. To understand how this wave is treated we consider the power spectrum, shown in Figure 8.8(c). We see amplitude at $k = 1, 2, 4$ as we would expect but now there is considerable amplitude at $k = 0$. The amplitude at $k = 1$ for the truth is also higher than we would expect. From Section 6.3 we know that this amplitude at $k = 0$ and $k = 1$ is caused by the long-wave. From Figure 8.8(c) it can be seen that the long-wave in the truth is also aliased to other wavenumbers when the DFT is performed over the LAM domain. This highlights an important distinction that we must be aware of. As well as understanding where information is aliased to so that we can investigate its effects, we also need to consider the magnitude of the aliasing effects, i.e. are the aliasing effects of the same magnitude in the analyses as they are in the truth. That is why we plot

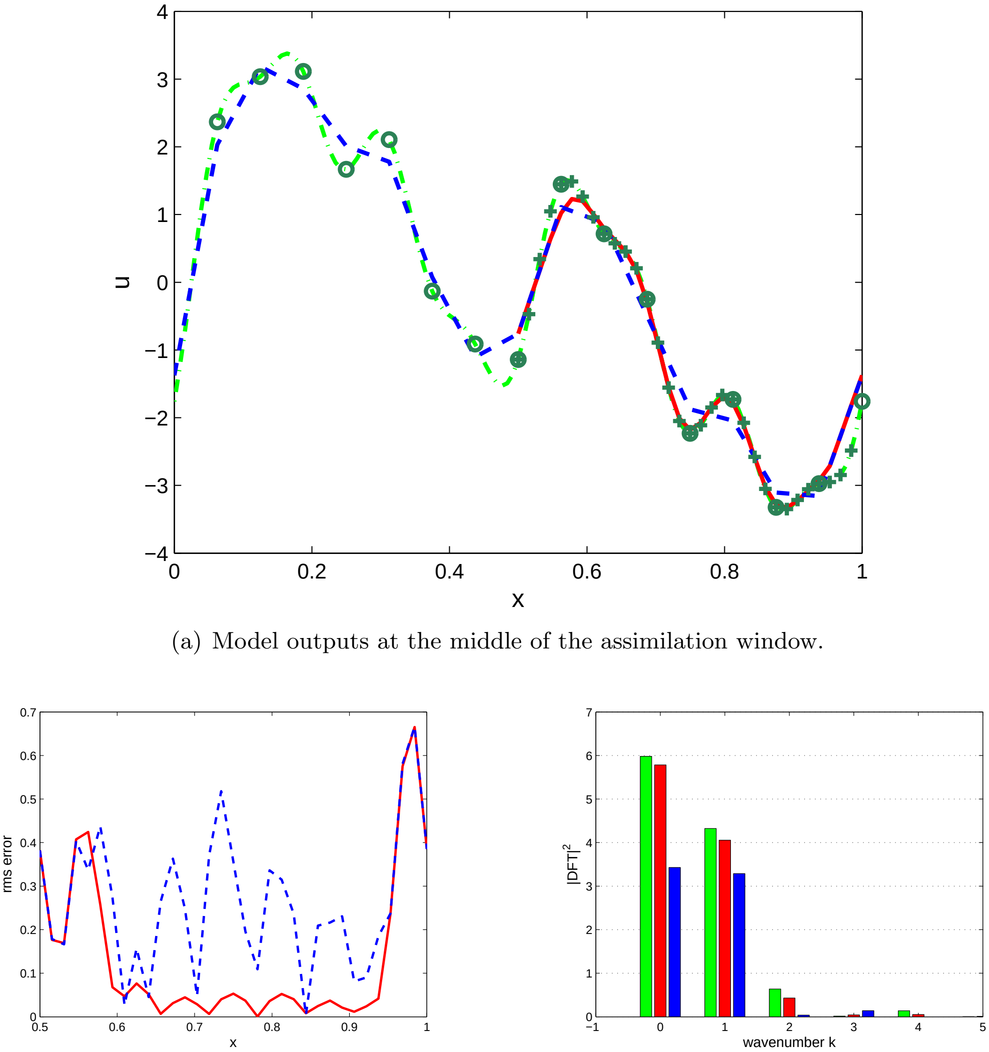


(a) Model outputs at $t = 0$.

(b) rms error between the truth and the model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

Figure 8.8: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2 x_j^r) + \sin(2\pi 4 x_j^r) + \sin(2\pi 8 x_j^r)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.



(a) Model outputs at the middle of the assimilation window.

(b) rms error between the truth and the model outputs at middle of the assimilation window.

(c) Power spectrum at the middle of the assimilation window.

Figure 8.9: The truth at $t = 0$ is $u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.

the power spectrum of the truth as well as the analyses. The comparison in magnitude between the truth and the analyses is as important as the actual amplitude location.

In Figure 8.8(c) we see that the LAM is capturing the aliased information at $k = 0$ well while the parent analysis does a poorer job. However, the parent analysis is more accurate at $k = 1$, and the inaccuracies at $k = 2$ have about the same magnitude in both analyses, but the LAM overestimates while the parent underestimates.

If we now consider the middle of the assimilation window, we see from Figure 8.9(a) that the LAM analysis again appears to fit the truth more closely than the parent. This is confirmed by the rms error shown in Figure 8.9(b). As before, the largest errors in the LAM are close to the boundaries, caused by errors introduced through the LBCs. Looking at the power spectrum shown in Figure 8.9(c) we see that at the middle of the assimilation window, the LAM matches the truth more closely at all wavenumbers.

From these examples we see that the long-wave information is aliased onto other wavenumbers, specifically the lowest wavenumbers that can be represented in our spectrum. A large proportion of the long-wave information is often aliased onto wavenumber $k = 0$. This is because of the properties of the discrete Fourier transform, as discussed in Section 6.3.2.

So far we have only considered a truth with a single long-wave. We now go on to consider a case with several long-waves to see if we can make the same conclusions.

8.2.2 Several long-waves

We have the reference trajectory

$$\begin{aligned} u^r(x^r_j) &= 2\sin(2\pi x^r_j/8) + 2\sin(2\pi x^r_j/2) + 2\sin(2\pi x^r_j) + 2\sin(2\pi 2x^r_j) \\ &\quad + \sin(2\pi 4x^r_j) + \sin(2\pi 8x^r_j) + \sin(2\pi 32x^r_j), \end{aligned}$$

at time $t = 0$. This contains several waves whose wavelengths are longer than the domain of the LAM. It also contains wavelengths that are too short to be resolved by the parent analysis as well as a wave too short to resolved by the LAM. The initial condition for the parent scheme is

$$u^P(x^P_i, 0) = \sin(2\pi x^P_i).$$

Figure 8.10(a) shows the truth and both analyses, in physical space, at the initial time $t = 0$. The LAM looks more accurate than the parent as it appears to be mapping the shape of the truth more closely. However, the shortest wave contained in the truth cannot be resolved, even by the LAM, and these high resolution oscillations in the truth make it difficult to judge the accuracy of the analyses by simply plotting their values. We therefore consider the rms error of both analyses, plotted in Figure 8.10(b). As can be seen, the LAM is significantly more accurate at all but a couple of isolated gridpoints, and as before, the largest errors in the LAM are close to the boundaries.

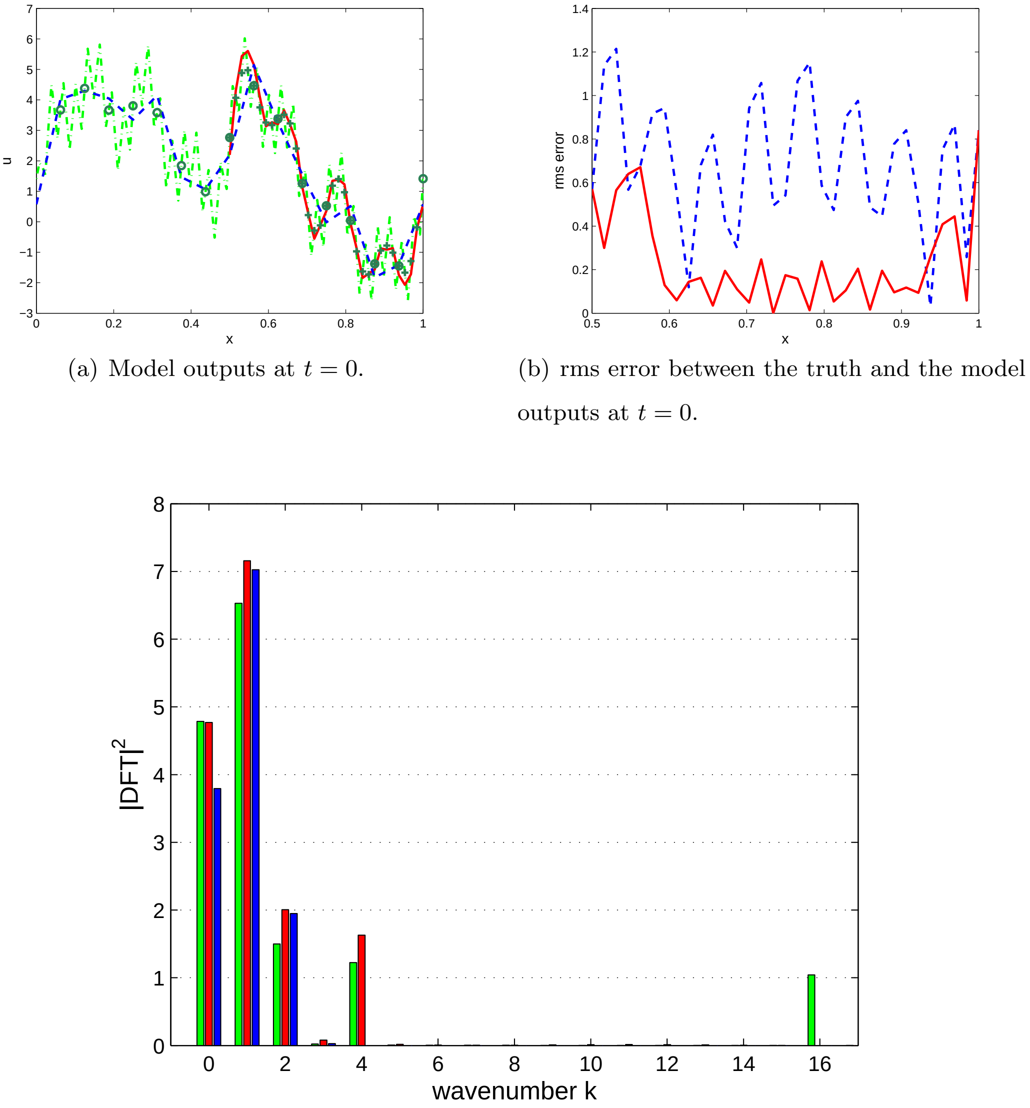
The accuracy of the analyses can also be understood by considering the power spectra. The power spectra of the truth and both analyses is shown in Figure 8.10(c). As we expect, the LAM captures wavenumbers missed by the parent. However, the magnitude at wavenumbers $k = 1$ and $k = 2$ is actually more accurate in the parent analysis.

As well as allowing us to compare the accuracy of the two analyses, the power spectrum also allows us to investigate the aliasing effect on the long-wave information. As in the one long-wave case, there is significant amplitude at $k = 0$. This is all aliased from the long-waves. There is also aliasing to other wavenumbers, in particular $k = 1$.

If we also consider the middle of the assimilation window, we see in Figure 8.11(a) that the LAM analysis still looks to fit the truth marginally more closely and this is confirmed by the rms error plotted in Figure 8.11(b). Figure 8.11(c) shows the power spectra of the truth and both analyses. The majority of the long-wave information is still being aliased to wavenumber $k = 0$ with some being sent to $k = 1$. By the middle of the assimilation window the parent analysis is significantly more accurate than the LAM at $k = 0$.

8.2.3 Summary of the long-wave results with a gridpoint scheme

In Chapter 6 we demonstrated that long-waves are aliased onto other wavenumbers. Here we have demonstrated that this same aliasing effect occurs with observational information of long-waves in the data assimilation. When the observations contain information from waves longer than the domain of the model, these long-waves are aliased onto other wavenumbers. The majority of the aliasing is onto the longest waves

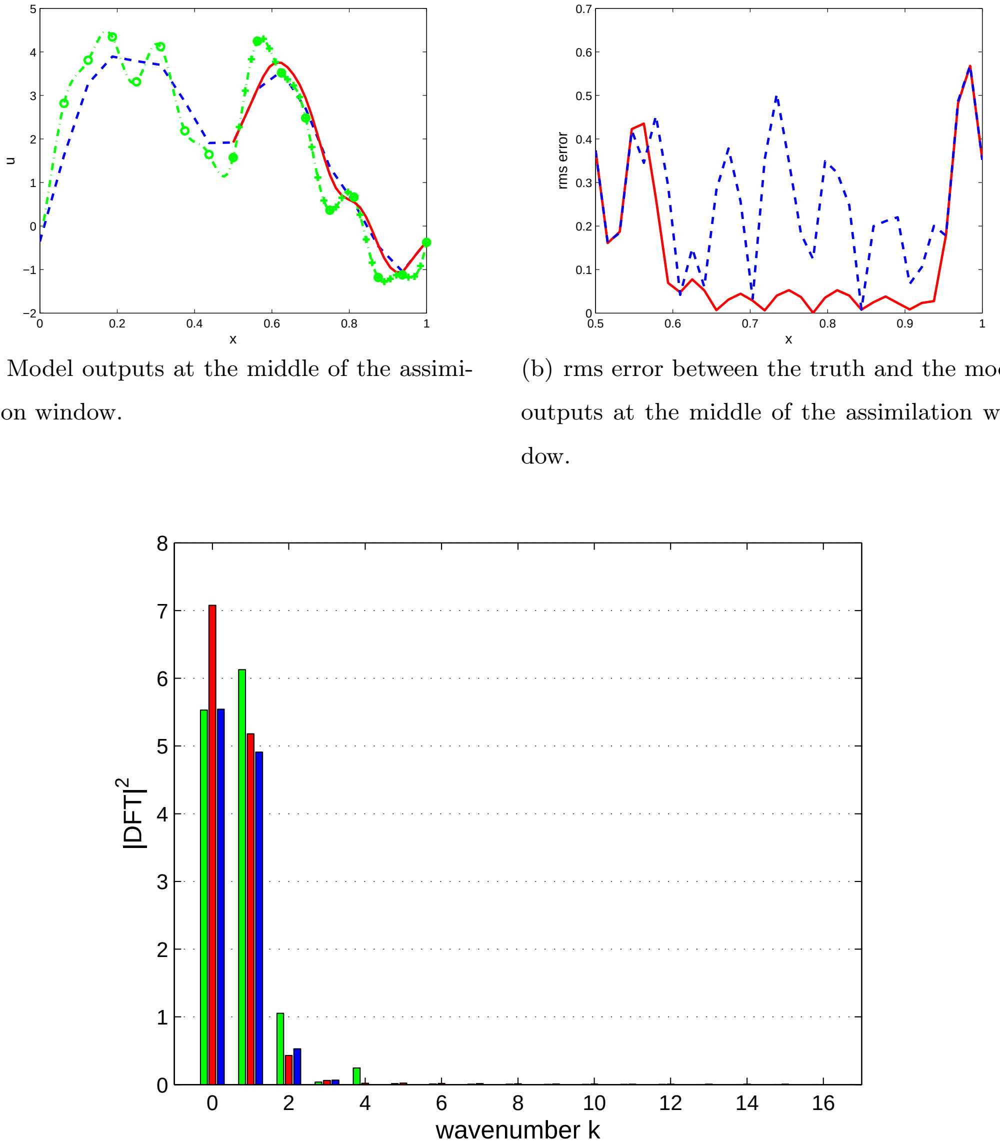


(a) Model outputs at $t = 0$.

(b) rms error between the truth and the model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

Figure 8.10: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r/8) + 2\sin(2\pi x_j^r/2) + 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r) + \sin(2\pi 32x_j^r)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.



(a) Model outputs at the middle of the assimilation window.

(b) rms error between the truth and the model outputs at the middle of the assimilation window.

(c) Power spectrum at the middle of the assimilation window.

Figure 8.11: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r/8) + 2\sin(2\pi x_j^r/2) + 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r) + \sin(2\pi 32x_j^r)$. In (a) the green dash-dotted line is the truth, the blue dash line is the parent analysis and the solid red line is the LAM analysis. The green circles are the observations seen by the parent data assimilation and the green crosses are the extra observations seen by the LAM data assimilation. In (b) blue is the error in the parent analysis and red is the error in the LAM analysis. In (c) green is the truth, red is the LAM analysis and blue is the parent analysis.

contained within the spectrum of the model. In particular, a significant amount of the long-wave information can be aliased onto wavenumber $k = 0$. We have demonstrated this for a single long-wave and a combination of several.

This aliasing is corrupting the longest waves on the LAM domain and we have shown that, in some cases, the parent analysis actually represents these wavenumbers more accurately than the LAM. This is because of the conflict in the LAM data assimilation between trying to fit to the high resolution observations while also being constrained to inaccurate boundary conditions. In Chapter 9 we go on to investigate whether this knowledge can be used to manipulate the data asimilation to generate a more accurate LAM analysis.

Aliasing long-wave information onto other wavenumbers could cause a problem for a spectral scheme. A spectral scheme that uses a sine transform will automatically set the value at $k = 0$ to be zero. This means that we potentially lose a significant proportion of the information about all the long-waves contained in a trajectory. The information aliased to other ($k \neq 0$) wavenumbers could also cause problems. The data assimilation scheme cannot distinguish between the information at $k = \kappa$ belonging to the wave with wavenumber κ and that coming from the aliased long-wave. As the data assimilation is being performed in spectral space this could cause the observational data to be corrupted and degrade the analysis. Also, because we are working in finite, discrete space the Fourier series is truncated. This loss of high resolution information could degrade the quality of the spectral analysis.

To investigate these potential problems we now repeat our experiments using the spectral scheme (Section 5.4.2). We compare the analysis generated by the spectral scheme with that generated by gridpoint scheme.

8.3 Long and short waves with the spectral scheme

To investigate the impact of the aliasing effect of the long-waves on a spectral scheme, we now compare the analysis generated by the spectral scheme (as defined in Section 5.4.2) with that generated by the gridpoint scheme for the same system.

Our spectral scheme uses a Fourier sine transform and therefore to compare the results in spectral space we now use the sine transform instead of the DFT, and it is this that is shown in the power spectra. Due to the use of the sine transform, information can no longer be aliased onto wavenumber $k = 0$, as it is in the DFT, as the sine transform automatically sets the amplitude to zero at $k = 0$. As well as comparing the analyses from the spectral and gridpoint schemes we therefore also need to investigate where the long-wave information is projected onto when we use a sine transform.

8.3.1 One long-wave

We start with the reference trajectory we had in Section 8.1

$$u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r),$$

at time $t = 0$.

We perform the LAM data assimilation in both spectral and gridpoint space. The LAM analysis shown in Figure 8.12(b) is that of the spectral scheme. The gridpoint analysis is not plotted because it lies directly under the spectral analysis. The exactness of this match is confirmed by the difference between the two, shown plotted in Figure 8.12(a). The magnitude of this difference is of $O(10^{-6})$ compared to the $O(1)$ magnitude of the analyses. The equivalence between the spectral and gridpoint assimilation is also confirmed as both analyses have identical power spectra, as shown in Figure 8.12(c).

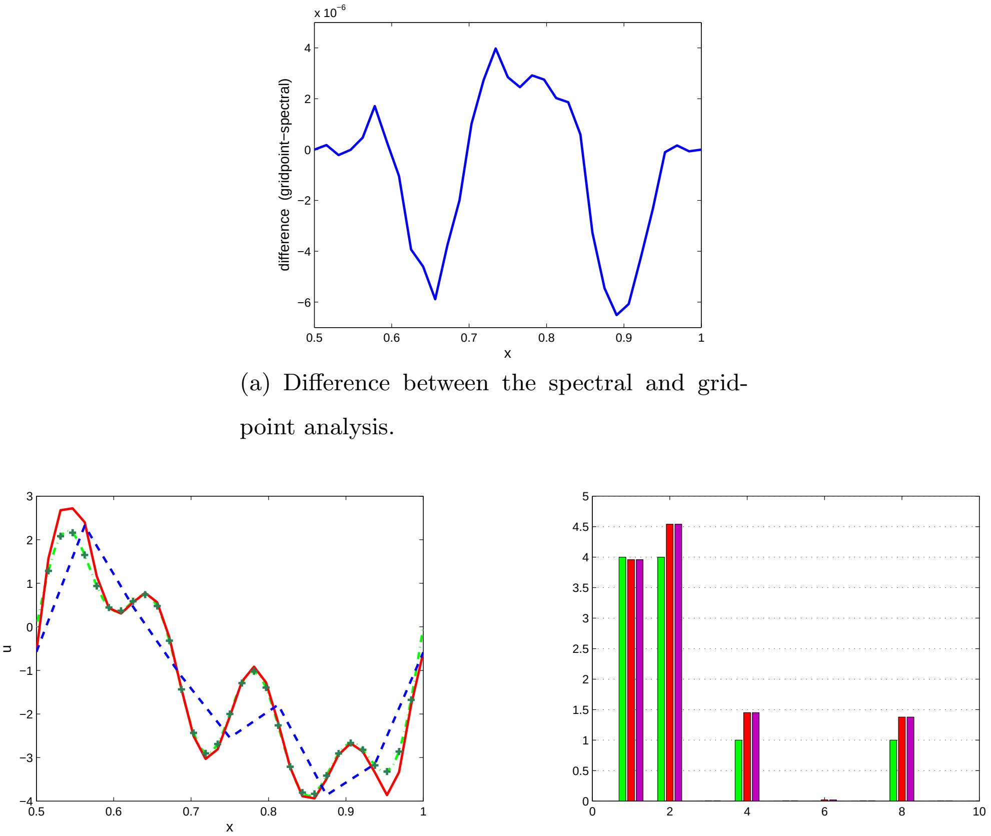
As discussed in Section 6.4, for the sine transform the wave with wavenumber $k = 1$ on the domain $L = 1/2$ is not treated as a long-wave. This is demonstrated in Figure 8.12(c) where the power spectrum of the truth has amplitude at $k = 1, 2, 4, 8$.

To consider the effect of a long wave with the spectral scheme we now consider the reference trajectory

$$u^r(x_j^r) = 2\sin(2\pi x_j^r/2) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r),$$

at time $t = 0$.

A LAM analysis is generated using both the spectral and gridpoint data assimilation schemes. To compare the two analyses, their difference is plotted in Figure 8.13(a). As

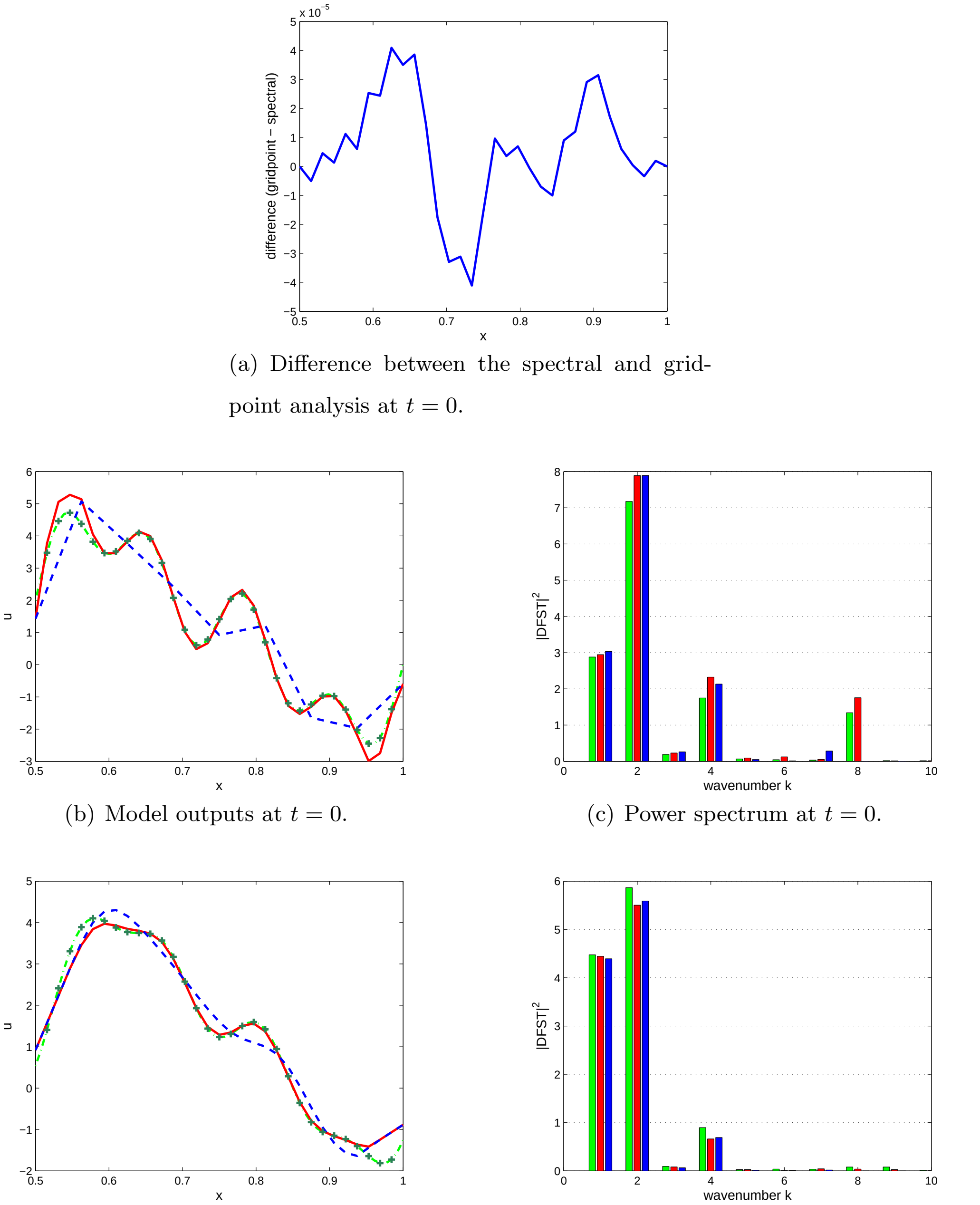


(a) Difference between the spectral and gridpoint analysis.

(b) Model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

Figure 8.12: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r)$ at $t = 0$. In (a) the green dash-dotted line is the truth, the red line is the LAM analysis, the blue dashed line is the background and the green crosses are the observations. In (c) green is the truth, red is the LAM analysis and pink is the gridpoint analysis.



(a) Difference between the spectral and grid-point analysis at $t = 0$.

(b) Model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

(d) Model outputs at the middle of the assimilation window.

(e) Power spectrum at the middle of the assimilation window.

Figure 8.13: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r/2) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r)$ at $t = 0$. In (b) and (d) the green dash-dotted line is the truth, the red line is the LAM analysis, the blue dashed line is the background and the green crosses are the observations. In (c) and (e) green is the truth, red is the LAM analysis and blue is the parent analysis.

can be seen, the two agree to $O(10^{-5})$. Therefore, in the other plots shown in Figure 8.13 only the spectral LAM analysis is plotted.

In Figures 8.13(b) and 8.13(d) the truth is plotted along with both the LAM and parent analyses, at the initial time and at the middle of the assimilation window respectively. In both we see that the LAM fits the truth more closely as it is able to pick out the higher resolution detail missed by the parent. This is confirmed by the power spectra plotted in Figures 8.13(c) and 8.13(e).

It is worth noting here that as occurred in Sections 8.1 and 8.2, although the LAM is more accurate in the sense that it picks out wavenumbers missed by the parent, at some wavenumbers the magnitude is actually more accurate in the parent than the LAM. For example, at $k = 4$ the parent is more accurate than the LAM at both times.

The power spectra shown in Figures 8.13(c) and 8.13(e) also allow us to examine how the long-wave is treated by the spectral scheme. In Section 8.2 we saw long-waves being aliased onto $k = 0$. However, the sine transform automatically sets $k = 0$ to be zero by definition. As can be seen in Figures 8.13(c) and 8.13(e), the long-waves are still aliased onto wavenumbers that are contained by the spectrum but a significant amount is now sent to $k = 1$, instead of $k = 0$ as before.

To see if this pattern of aliasing applies when we have more than one long-wave, we now consider a case with several long-waves.

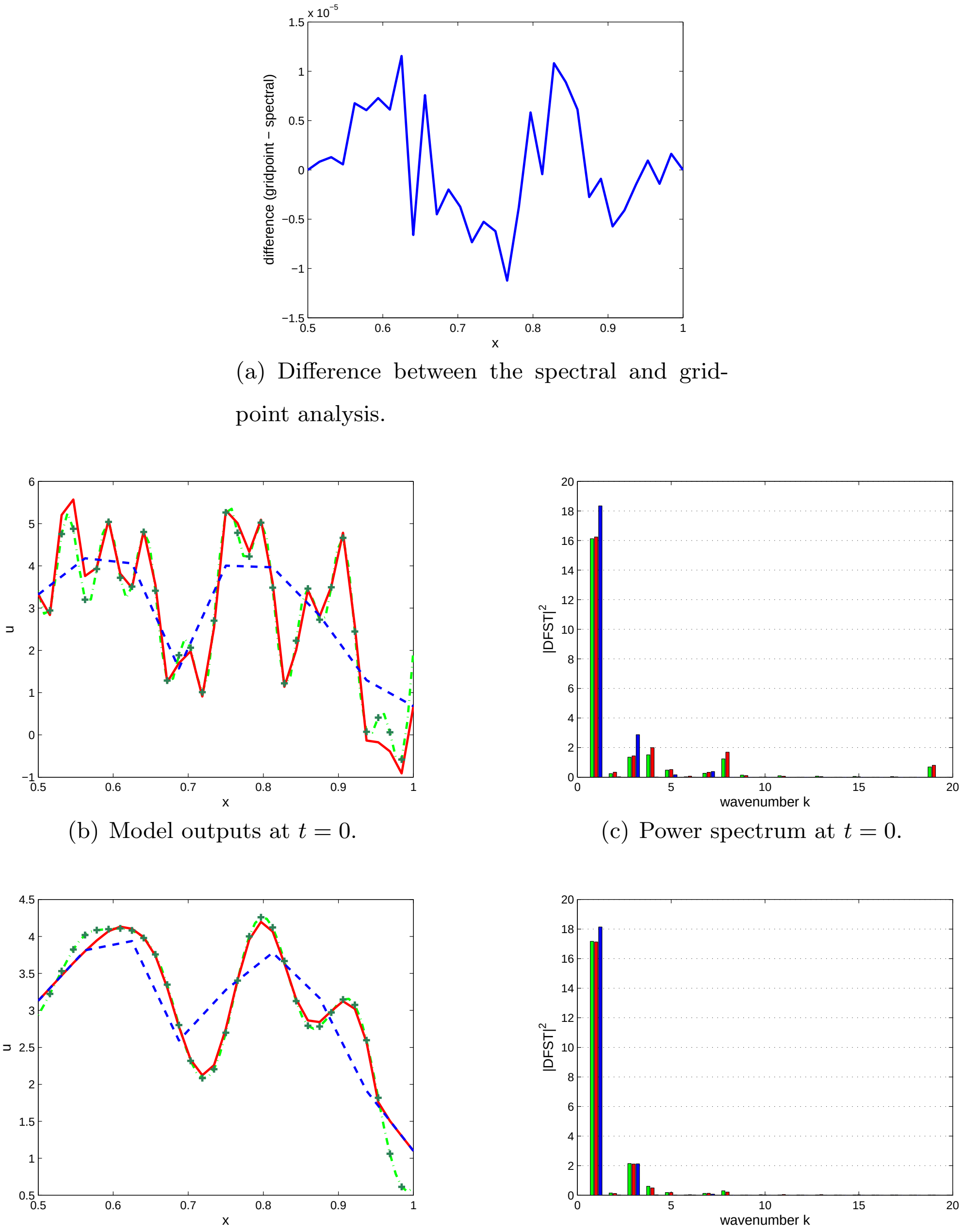
8.3.2 Several long-waves

We now consider the reference trajectory

$$\begin{aligned} u^r(x_j^r) &= 2\sin(2\pi x_j^r/4) + 2\sin(2\pi x_j^r/2) + \sin(2\pi 4x_j^r) \\ &\quad + \sin(2\pi 7x_j^r) + \sin(2\pi 8x_j^r) + + \sin(2\pi 19x_j^r), \end{aligned}$$

at time $t = 0$.

Figure 8.14(a) shows the difference between the spectral and gridpoint LAM analyses. As with the single long-wave case, Section 8.3.1, the difference is of $O(10^{-5})$ and so we only plot the spectral LAM analysis in the other plots in Figure 8.14.



(a) Difference between the spectral and gridpoint analysis.

(b) Model outputs at $t = 0$.

(c) Power spectrum at $t = 0$.

(d) Model outputs at the middle of the assimilation window.

(e) Power spectrum at the middle of the assimilation window.

Figure 8.14: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r/4) + 2\sin(2\pi x_j^r/2) + \sin(2\pi 4x_j^r) + \sin(2\pi 7x_j^r) + \sin(2\pi 8x_j^r) + \sin(2\pi 19x_j^r)$ at $t = 0$. In (b) and (d) the green dash-dotted line is the truth, the red line is the LAM analysis, the blue dashed line is the background and the green crosses are the observations. In (c) and (e) green is the truth, red is the LAM analysis and blue is the parent analysis.

Again, like in Section 8.3.1, in both Figures 8.14(b) and 8.14(d) we see that at both the initial time and the middle of the assimilation window respectively, the LAM analysis fits the truth more closely as it can pick out the higher resolution detail lost by the parent.

Considering the power spectra shown in Figures 8.14(c) and 8.14(e) we see that the long-waves are sent to wavenumber $k = 1$. Wavenumber $k = 1$ contains the information from several different long-waves. However, this does not cause problems for the data assimilation as evidenced by the agreement between the spectral and gridpoint LAM analysis.

As well as considering sine wave examples, we also need to consider other waveforms. In particular, we should consider waveforms with a slowly converging Fourier series, to see if this causes a problem for the spectral scheme.

8.3.3 Non-sine waveforms with the spectral assimilation scheme

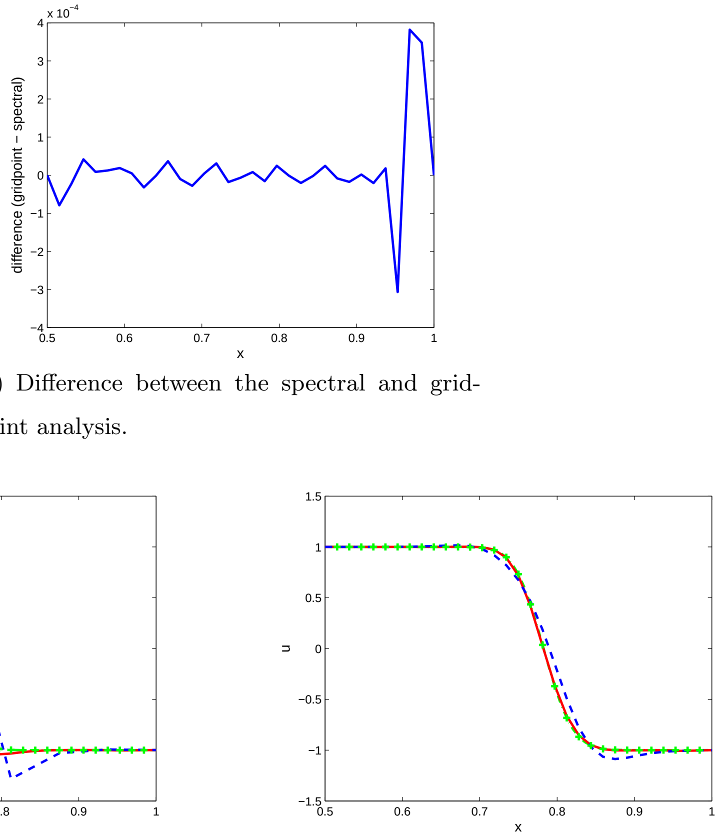
We now consider a tanh wave example, as this has a slowly converging Fourier series. The truncating of higher order terms by the discrete sine transform may therefore cause the spectral scheme to lose information and produce a poorer analysis.

We have the reference trajectory

$$\begin{aligned} u^r(x^r_j) &= \tanh(64x^r_j - 16) \qquad \text{for } j = 1, \ldots, S/2 \\ u^r(x^r_{S+1-j}) &= \tanh(64x^r_j - 16) \qquad \text{for } j = 1, \ldots, S/2 \end{aligned} \tag{8.1}$$

at time $t = 0$, where S is the number of gridpoints in the reference trajectory.

The model outputs in physical space are plotted in Figure 8.15. Figure 8.15(a) shows the difference between the gridpoint and spectral scheme LAM analyses. As we see, the largest difference between them is of $O(10^{-4})$ compared to the $O(1)$ magnitude of the analyses. This illustrates that the spectral and gridpoint schemes produce the same analysis, even for functions with slowly converginging Fourier series. Given this high level of agreement, just the spectral scheme analysis is plotted in Figures 8.15(b) and 8.15(c).



(a) Difference between the spectral and gridpoint analysis.

(b) Model outputs at $t = 0$.

(c) Model outputs at the middle of the assimilation window.

Figure 8.15: The truth is equation (8.1) at $t = 0$. The green dash-dotted line is the truth, the red line is the LAM analysis, the blue dashed line is the background and the green crosses are the observations.

Figure 8.15(b) shows the model outputs at $t = 0$. We see that the parent analysis over-compensates in the regions of rapid change. Although this has diminished by the middle of the assimilation window, it is still evident. At both times the LAM analysis is more accurate accross the entire LAM domain.

8.3.4 Summary of the spectral scheme results

When using a Fourier sine transform over a LAM domain, long-waves are aliased onto other wavenumbers. The majority goes to the longest waves contained in the spectrum. In particular, a significant amount goes to $k = 1$. However, in contrast to Section 8.2 where we used the DFT, $k = 0$ is always set to zero, due to the definition of the sine transform.

We also see that although the long-waves are aliased onto other wavenumbers, this does not cause a problem for the spectral data assimilation scheme in our model. The spectral scheme produced the same analysis as that produced by the gridpoint scheme, to at least $O(10^{-5})$ accuracy.

The agreement between the spectral and gridpoint schemes is because of the discrete nature of the schemes. If we truncate the continuous Fourier sine transform at wavenumber $k = N$, then information contained at wavenumbers higher than N would be lost. However, the discrete Fourier sine transform does not suffer in the same way. If we have a finite interval divided into N equally spaced points then the discrete sine transform over the interval is automatically truncated at $k = N$ by definition. However, as discussed in Section 4.2.2, on N gridpoints we cannot represent waves with wavenumbers higher than N. Any waves with wavenumbers higher than N are aliased onto longer waves. Therefore, no information is lost by the truncation in the discrete sine transform. This can be demonstrated by comparing a function f_j and what we get back if we transform into spectral space and then transform back.

From Section 5.4.2, the sine transform of a function f_j is

$$\widetilde{f}_k = \sqrt{\frac{2}{N}} \sum_{j=1}^{N-1} f_j \sin\left(\pi jk/(N)\right).$$

If we now transform $\widetilde{f}_k$ back into physical space we get

$$g_j = \sqrt{\frac{2}{N}} \sum_{k=1}^{N-1} \left[\sqrt{\frac{2}{N}} \sum_{j=1}^{N-1} f_j \sin\left(\pi jk/(N)\right) \right] \sin\left(\pi jk/(N)\right).$$

If information is being lost by the transform then we would expect $f_j \neq g_j$.

$$\begin{aligned} g_j &= \frac{2}{N} \sum_{j=1}^{N-1} f_j \sum_{k=1}^{N-1} \sin^2\left(\pi jk/(N)\right), \\ &= \sum_{j=1}^{N-1} f_j \delta_{j,k}, \quad \text{by orthogonality,} \\ &= f_j. \end{aligned}$$

i.e. the truncation is not causing high resolution information to be lost by the transformation.

8.4 The background trajectory $\boldsymbol{x_i^b}$

How we generate the background trajectory $\boldsymbol{x_i^b}$ can have a significant effect on the LAM 4D-Var analysis.

It was stated in Section 5.3 that the model in incremental space is equivalent to the model in full state space, if we have a linear model. This equivalence is also true for the cost function, since

$$(\boldsymbol{y_i} - \boldsymbol{H x_i}) = \left(\boldsymbol{y_i} - \boldsymbol{H x_i^b} - \boldsymbol{H \delta x_i}\right) = (\boldsymbol{\delta y_i} - \boldsymbol{H \delta x_i}), \tag{8.2}$$

where $\boldsymbol{\delta y_i} = \boldsymbol{y_i} - \boldsymbol{H x_i^b}$ are the innovations and $\boldsymbol{x_i^b}$ is the background trajectory. However, there is an error that can be introduced into equation (8.2) depending on how we generate the background trajectory $\boldsymbol{x_i^b}$.

To maintain the equivalence between the full space and the incremental space shown in equation (8.2), the background trajectory for the LAM should be the background value at time t_0 interpolated to the LAM grid, and then the LAM model run from this initial condition to provide the background trajectory values $\boldsymbol{x_i^b}$ to be used in the calculation of the innovations $\boldsymbol{\delta y_i}$. Symbolically, this can written as

$$\boldsymbol{x_0^{bP}} \quad \overset{\text{interpolate}}{\longrightarrow} \quad \boldsymbol{x_0^b} \quad \overset{\boldsymbol{x_i^b = M x_{i-1}^b}}{\longrightarrow} \quad \left\{\boldsymbol{x_i^b}\right\}_{i=0}^{\mathcal{T}},$$

where $\boldsymbol{x_0^{bP}}$ is the parent resolution background at time t_0 and $\boldsymbol{x_0^b}$ is the LAM resolution background at time t_0.

However, the background trajectory values $\boldsymbol{x_i^b}$ could instead be generated by simply interpolating the parent model values at each timestep.

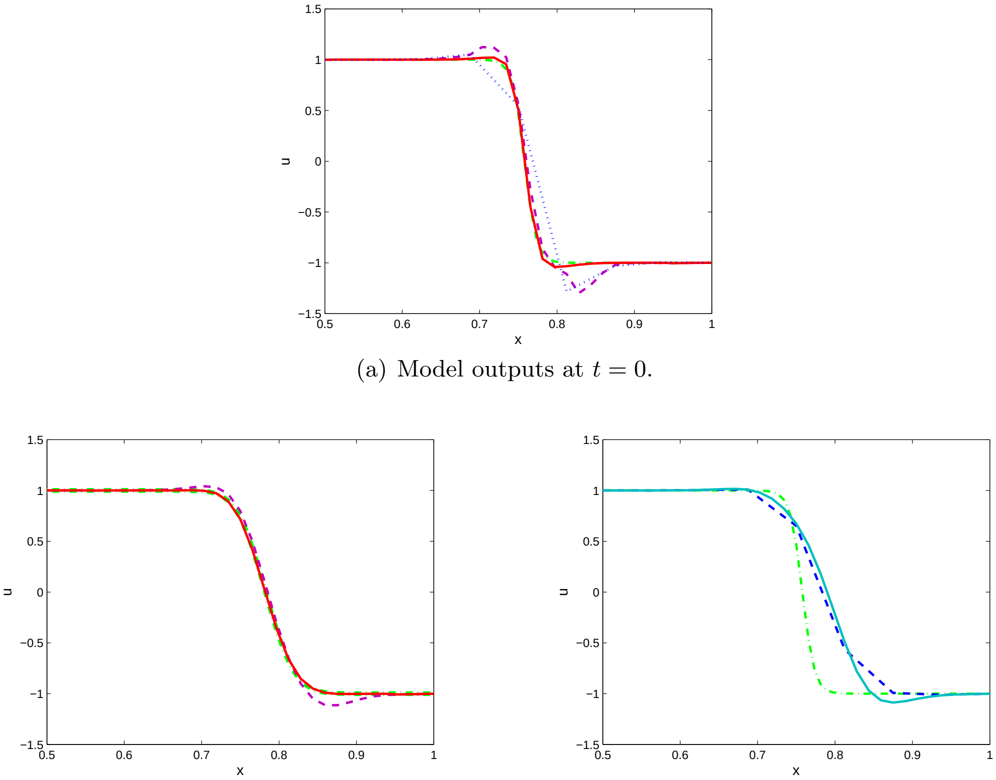
$$\left\{\boldsymbol{x_i^{bP}}\right\}_{i=0}^{\mathcal{T}} \quad \overset{\text{interpolate}}{\longrightarrow} \quad \left\{\boldsymbol{x_i^b}\right\}_{i=0}^{\mathcal{T}},$$

This results in different background trajectory values $\boldsymbol{x_i^b}$, due to the fact that running the model at different resolutions can yield different values, even at the common gridpoints. Running the model at a low resolution then interpolating does not yield the same result as running the model at a high resolution.

This difference in the background trajectories is demonstrated in Figure 8.16(c) for the tanh wave example we saw in Section 8.3.3. It shows the values to be used for $\boldsymbol{x_i^b}$ if we interpolate a low resolution model run (call this background 1) compared to the values if we run the model at a higher resolution (call this background 2). As we see in Figure 8.16(c), there is a clear difference in the two choices.

If we now consider the analysis generated using each choice of background, we see that each choice can result in a very different analysis. Figure 8.16(a) shows the two analyses at the initial time $t = 0$. Only one background is shown because both the backgrounds have the same initial values. As we can see, the analysis generated with background 1 deviates more from the truth. This is caused by the error introduced by the incorrect conversion to incremental space. As well as being present at the initial time, if we consider Figure 8.16(b), which shows the analyses at the middle of the assimilation window, we see that this difference between the two analyses is still present and the analysis generated with background 1 is still less accurate.

The extent of the difference made on the analysis by the choice of background depends on the form of the wave being captured by the analysis. On the tanh wave shown in Figure 8.16 the difference is easily visable, and if we consider a background with a phase shift, as shown in Figure 8.17, the difference is even more apparent. However, for the sine wave examples the difference is much less noticable, as illustrated in Figure 8.18. This difference in dependence is caused by how variable the model output is with different resolutions. Figure 8.19 shows the difference in model output between a resolution of

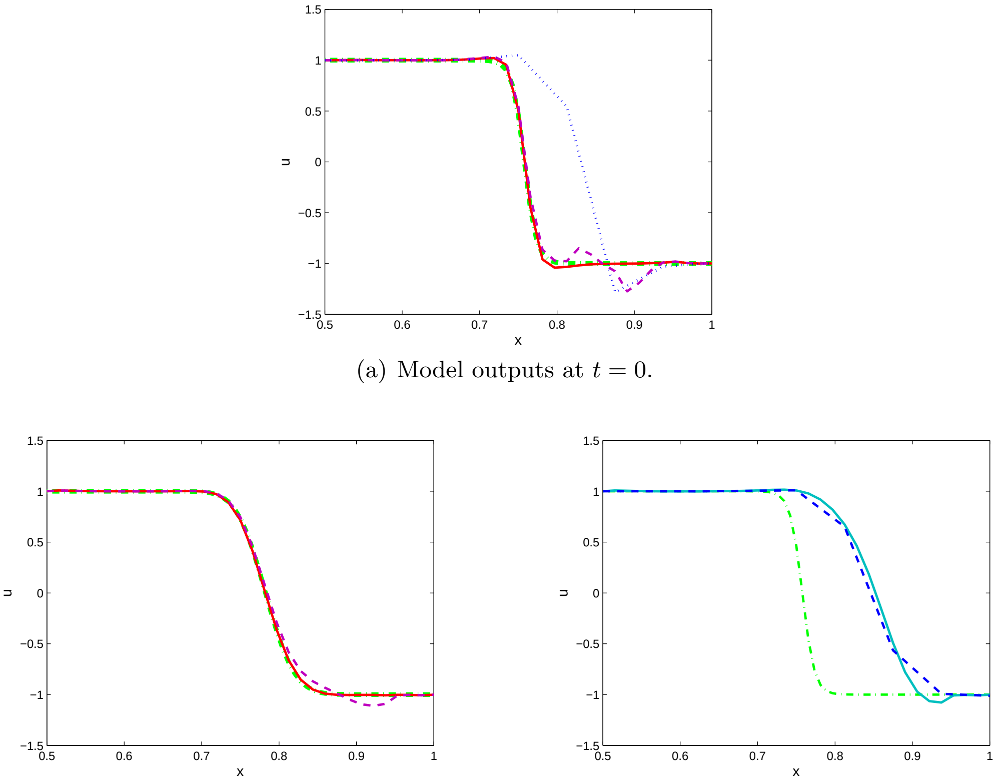


(a) Model outputs at $t = 0$.

(b) Model outputs at the middle of the assimilation window.

(c) Background values at the middle of the assimilation window.

Figure 8.16: The truth is given by equation (8.1) at $t = 0$. In (a) and (b) the green dash-dotted line is the truth, the red line is the LAM analysis with backgroud 2, the pink dashed line is the LAM analysis with background 1 and only in (a) the blue dotted line is the background. In (c) green is the truth, the blue dash line is background 1 and the solid pale blue line is background 2.

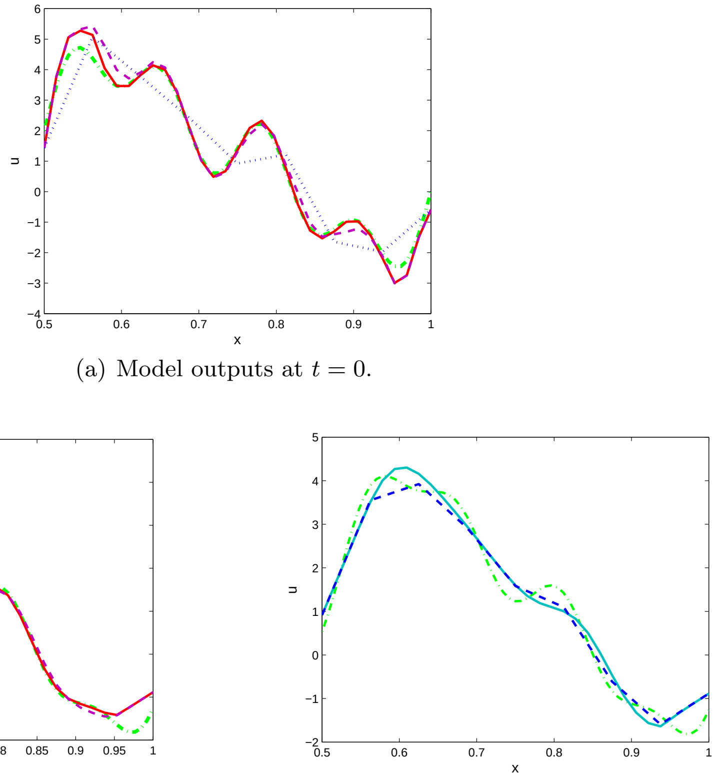


(a) Model outputs at $t = 0$.

(b) Model outputs at the middle of the assimilation window.

(c) Background values at the middle of the assimilation window.

Figure 8.17: The truth is given by equation (8.1) at $t = 0$ and the background has a phase shift. In (a) and (b) the green dash-dotted line is the truth, the pink line is the LAM analysis with background 1, the red line is the LAM analysis with backgroud 2 and in (a) only the blue dotted line is the background. In (c) green is the truth, the blue dash line is background 1 and the solid pale blue line is background 2.



(a) Model outputs at $t = 0$.

(b) Model outputs at the middle of the assimilation window.

(c) Background values at the middle of the assimilation window.

Figure 8.18: The truth is $u^r(x_j^r) = 2\sin(2\pi x_j^r) + 2\sin(2\pi 2x_j^r) + \sin(2\pi 4x_j^r) + \sin(2\pi 8x_j^r)$ at $t = 0$. In (a) and (b) the green dash-dotted line is the truth, the pink line is the LAM analysis with background 1, the red line is the LAM analysis with backgroud 2 and in (a) only the blue dotted line is the background. In (c) green is the truth, the blue dash line is background 1 and the solid pale blue line is background 2.

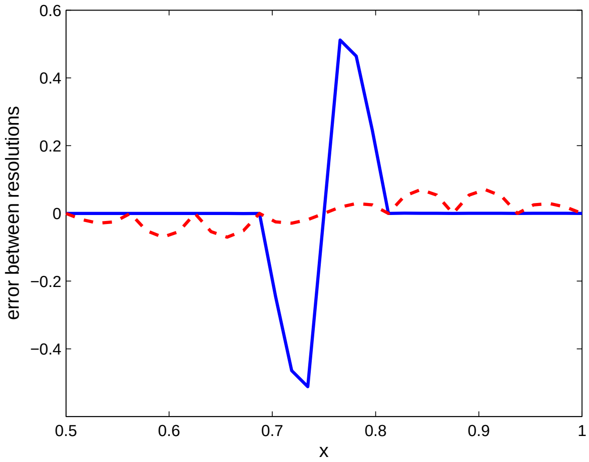


Figure 8.19: Error between model outputs with different resoltions for sine and tanh waves.

16 and 64 gridpoints, for both a tanh and a sine wave. As can be seen, the sine wave has a small difference throughout the domain whereas the tanh wave agrees at all but a few gridpoints. However, when the tanh wave does differ it differs by a much more significant amount, 5 times that of the sine wave. It is this large inaccuracy that causes the noticable differences seen in the analysis.

From these examples we see that it is important to generate the background values from a model run at the same resolution as that of the assimilation to be carried out. However, with models being run at a very high resolution, this extra run of the high resolution model to produce the background values is an added expense on already stretched computer resources. However, if the LAM is being cycled and the background is coming from a previous LAM forecast, with only the boundary couditions being provided by the parent model, then the background is already at the higher resolution and there is no longer an issue. This cycling on the LAM, with a previous LAM forecast being used to provide the LAM background, is what is done operationally at the UK Met Office [3].

8.5 Summary

We have shown that, as we would expect, running a 4D-Var scheme at a higher resolution does enable the analysis to capture the smaller scales which are missed by the parent analysis. However, we also showed that to do this the LAM data assimilation requires high resolution observations.

We also note however, that even with perfect observations and no background term in the cost function, the LAM analysis cannot correctly reconstruct the truth, due to the influence of the background trajectory at the LBCs and buffer zone. We have also shown that these errors caused by the LBCs affect the low wavenumbers of the LAM spectrum.

Due to the model being less diffusive at higher resolution the LAM analysis also has errors introduced by trying to compensate for the difference in diffusion rate compared to the truth. These errors occur on all wavenumbers but the high wavenumbers are affected to a lesser degree due to these scales diffusing away more rapidly.

When the observations contain information from waves longer than the domain of the model, these long-waves are aliased onto other wavenumbers. The majority of the aliasing is onto the longest waves contained within the spectrum of the model. In particular, when using the DFT a significant amount of the long-wave information is aliased onto wavenumber $k = 0$. In contrast, when using the Fourier sine transform $k = 0$ is always set to zero and instead a significant amount of the long-wave information is aliased onto wavenumber $k = 1$. This has been demonstrated for a single long-wave and a combination of several.

The LBCs and aliasing of long-wave information is corrupting the low wavenumbers in the LAM analysis and we have shown that, in some cases, the parent analysis actually represents these wavenumbers more accurately. In Chapter 9 we go on to investigate whether this knowledge can be used to manipulate the data asimilation to generate a more accurate LAM analysis.

Although the long-waves are aliased onto other wavenumbers, we have shown that this does not cause a problem for the spectral data assimilation scheme. The spectral scheme produced the same analysis as that produced by the gridpoint scheme, to at least $O(10^{-5})$ accuracy. This agreement between the spectral and gridpoint schemes is because of the discrete nature of the schemes.

We have also shown the importance of generating the background values from a model run at the same resolution as that of the assimilation to be carried out. Simply interpolating a coarser model run introduces errors into the LAM analysis. However, it is noted that if the LAM is being cycled and the background is coming from a previous LAM forecast, with only the boundary counditions being provided by the parent model, then the background is already at the higher resolution and there is no longer an issue.

Throughout this Chapter the background error covariance matrix has been set to zero. We now go on to consider the spectral scheme with a non-zero background error covariance matrix. We investigate whether we can control how the data assimilation affects different wavenumbers by manipulating the background error covariance matrix in spectral space.

Chapter 9

Manipulating scales in a 4D-Var analysis using the background error covariance matrix Σ

Now that we understand where information about different scales is projected in a 4D-Var scheme, we want to see if we can use this knowledge to improve the LAM analysis. We want to improve the representation of low wavenumbers because these wavenumbers are inversely affected by the LBCs, as discussed in Section 8.1.2, and because it is the low wavenumbers that contain the aliased long-wave information, as discussed in Section 8.2. We aim to improve the accuracy of the low wavenumbers whilst maintaining the improvement in the small scales achieved by the increase in resolution of the LAM, compared to the parent model.

In Chapter 8 we always had the inverse background error covariance matrix set to be zero. We therefore had no background term J_b in the cost function. We now introduce a non-zero inverse background error covariance matrix and aim to use this matrix to constrain the different scales.

As discussed in Section 5.4.2, in the spectral scheme we use a control variable transform (the sine transform) and write the cost function (equation (5.6)) in terms of a new control variable $\boldsymbol{\delta z}$, where $\boldsymbol{\delta z}$ is related to $\boldsymbol{\delta x}$ by equation (5.9),

$$\boldsymbol{\delta z} = \boldsymbol{W\delta x},$$

where $\boldsymbol{W}$ is the sine transform. The cost function is now given by equation (5.13),

$$J(\boldsymbol{z_0}) = \frac{1}{2}\boldsymbol{\delta z_0}^T\boldsymbol{\Sigma^{-1}\delta z_0} + \frac{1}{2}\sum_{k=0}^{\mathcal{T}}(\boldsymbol{\delta y_k} - \boldsymbol{HU\delta z_k})^T\boldsymbol{R_k^{-1}}(\boldsymbol{\delta y_k} - \boldsymbol{HU\delta z_k}),$$

where $\boldsymbol{U}$ is the inverse sine transform and $\boldsymbol{\Sigma^{-1}}$ is given by equation (5.14),

$$\boldsymbol{\Sigma^{-1}} = \boldsymbol{U^T B^{-1} U}.$$

From equation (5.13) we see that, for the spectral scheme, we can consider the background error covariance matrix in physical space, matrix $\boldsymbol{B}$, or in spectral space, matrix $\boldsymbol{\Sigma}$.

The background error covariance matrix controls how much the observational information is spread out. In physical space the spread of observational information is based on proximity to the observation. The matrix $\boldsymbol{B}$ controls how far an observation's influence will reach (radius of influence) and by how much. The matrix $\boldsymbol{B}$ is defined by how one point in space is connected to another. Its covariances depend on physical length scales. In contrast, the matrix $\boldsymbol{\Sigma}$ is defined in terms of wavenumbers.

The diagonal entries of an error covariance matrix are the variances. In spectral space they are the variances on the different wavenumbers. We aim to use the knowledge about the different wavenumbers gained in Chapter 8 to allow us to choose the diagonal entries of matrix $\boldsymbol{\Sigma}$ in such a way as to control how the different wavenumbers are affected by the data assimilation scheme. We then aim to use this control of how the different wavenumbers are treated to enable the LAM analysis to accurately capture all the scales present in the truth.

We begin by comparing, in physical space, the matrix $\boldsymbol{B}$ corresponding to choices of matrix $\boldsymbol{\Sigma}$.

9.1 How the matrix $\boldsymbol{B}$ corresponds to choices of the matrix $\boldsymbol{\Sigma}$

While it is useful for us to define the background error covariance matrix in spectral space, as this allows us to specify the variances on the different wavenumbers instead of physical locations, it is important to consider how the choice of the matrix $\boldsymbol{\Sigma}$ relates to the corresponding matrix $\boldsymbol{B}$.

We consider $\boldsymbol{\Sigma_\gamma} = \text{diag}\{\gamma_i\}$ and $\boldsymbol{\Sigma_\mu} = \text{diag}\{\mu_i\}$ for a system with 31 gridpoints where

$$\gamma_i = \begin{cases} 0.005 & \text{if } i = 1,2,3 \\ 0.01 & \text{if } i = 4,\ldots,10 \\ 0.1 & \text{if } i = 11,\ldots,16 \\ 0.5 & \text{if } i = 17,\ldots,24 \\ 1 & \text{if } i = 25,\ldots,31 \end{cases} \tag{9.1}$$

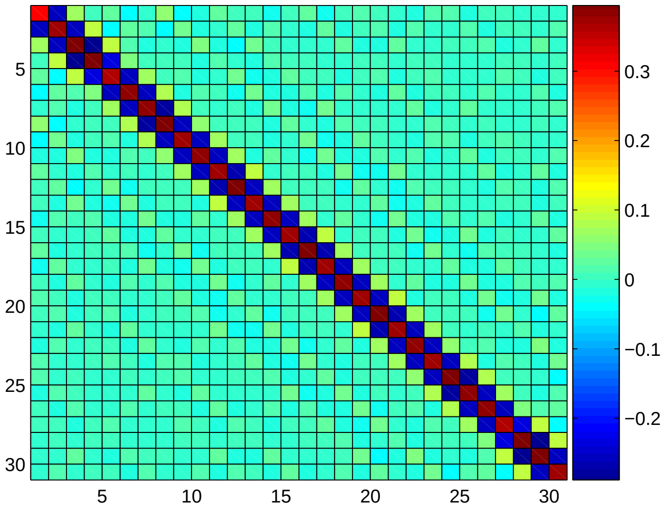
and

$$\mu_i = \begin{cases} 1 & \text{if } i = 1,2,3 \\ 0.5 & \text{if } i = 4,\ldots,10 \\ 0.1 & \text{if } i = 11,\ldots,16 \\ 0.01 & \text{if } i = 17,\ldots,24 \\ 0.005 & \text{if } i = 25,\ldots,31\ . \end{cases} \tag{9.2}$$

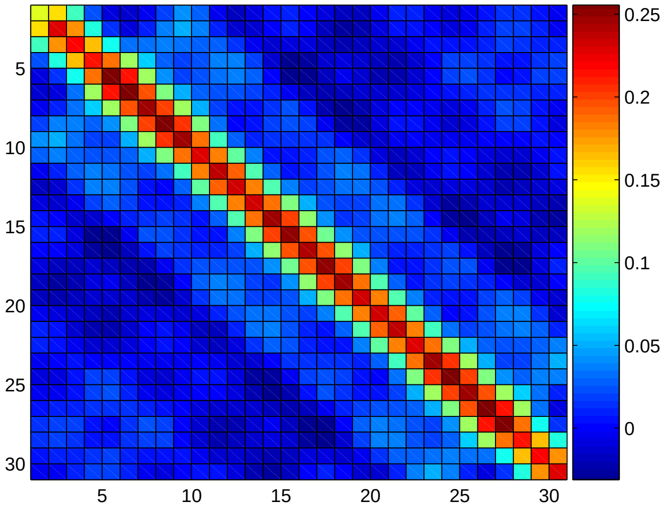
In $\boldsymbol{\Sigma_\gamma}$ we have placed a very small variance on the large scales and a large variance on the small scales. This corresponds to what we would expect to do in our model, constraining the large scales to match the background while allowing the observations to influence the smaller scales. As can be seen in Figure 9.1(a), the corresponding matrix $\boldsymbol{B_\gamma}$ displays an oscillating structure, suggesting that the correlations are in the shortest wavelengths, as we would hope. This is a good sign and warrants further investigation.

In $\boldsymbol{\Sigma_\mu}$ we have placed a large variance on the large scales and a small variance on the small scales. This results in the corresponding matrix $\boldsymbol{B_\mu}$ resembling a more traditional matrix $\boldsymbol{B}$ with correlations on the large scales, as can be seen in Figure 9.1(b).

This can be further seen by considering a single row of the matrix $\boldsymbol{B}$. This is known as the structure function and illustrates how information is spread by the matrix $\boldsymbol{B}$. The



(a) Plot of matrix $\boldsymbol{B}_{\gamma}$.



(b) Plot of matrix $\boldsymbol{B}_{\mu}$.

Figure 9.1: Plots of the matrix $\boldsymbol{B}$.

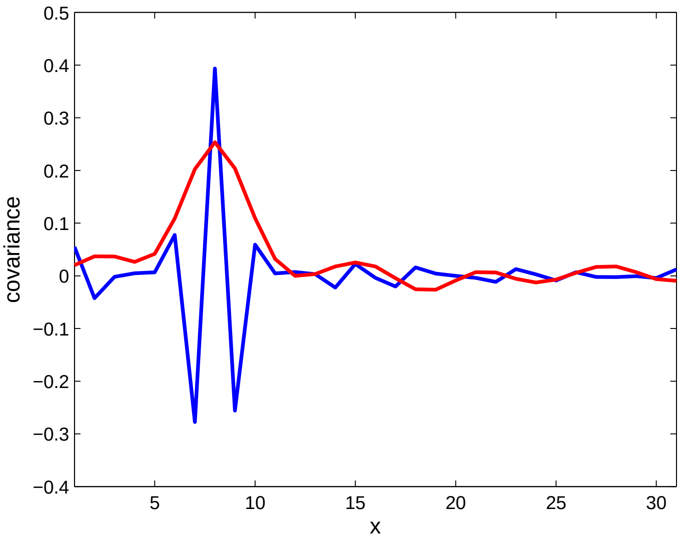


Figure 9.2: Plot of row 8 of the matrices $\boldsymbol{B}_\gamma$, in blue, and $\boldsymbol{B}_\mu$, in red.

structure function of matrix $\boldsymbol{B_\gamma}$ and matrix $\boldsymbol{B_\mu}$ are plotted in Figure 9.2. As we see, the blue line corresponding to $\boldsymbol{B_\gamma}$ oscillates rapidly which will result in increments on the small scales. In comparison, the red line corresponding to $\boldsymbol{B_\mu}$ is smoother with a wider spread which will result in increments on the larger scales. These are promising results as they suggest that we can alter which scales are affected in the LAM analysis by our choice of matrix $\boldsymbol{\Sigma}$. It is therefore worth testing different choices of $\boldsymbol{\Sigma}$ in the model. Initial tests with $\boldsymbol{\Sigma^{-1}} \neq 0$ are shown in Section 9.2.

9.2 Initial tests with a non-zero background error covariance matrix $\boldsymbol{\Sigma}$

We begin by investigating whether, by our choice of matrix $\boldsymbol{\Sigma}$, we can control which wavenumbers are affected by the data assimilation.

We first run the 4D-Var spectral scheme with $\boldsymbol{\Sigma_0^{-1}} = \boldsymbol{0}$ (no background term in the cost function), $\boldsymbol{\Sigma_1^{-1}} = \boldsymbol{I}$ (a small variance on all scales), $\boldsymbol{\Sigma_\alpha^{-1}} = \mathrm{diag}\{\alpha_i\}$ and $\boldsymbol{\Sigma_\beta^{-1}} = \mathrm{diag}\{\beta_i\}$, where

$$\alpha_i = \begin{cases} 0 & \text{if } i < M/2 \\ 1 & \text{if } i \geq M/2 \; , \end{cases}$$

(no background term for the large scales and a small variance on the small scales) and

$$\beta_i = \begin{cases} 1 & \text{if } i < M/2 \\ 0 & \text{if } i \geq M/2 \ , \end{cases}$$

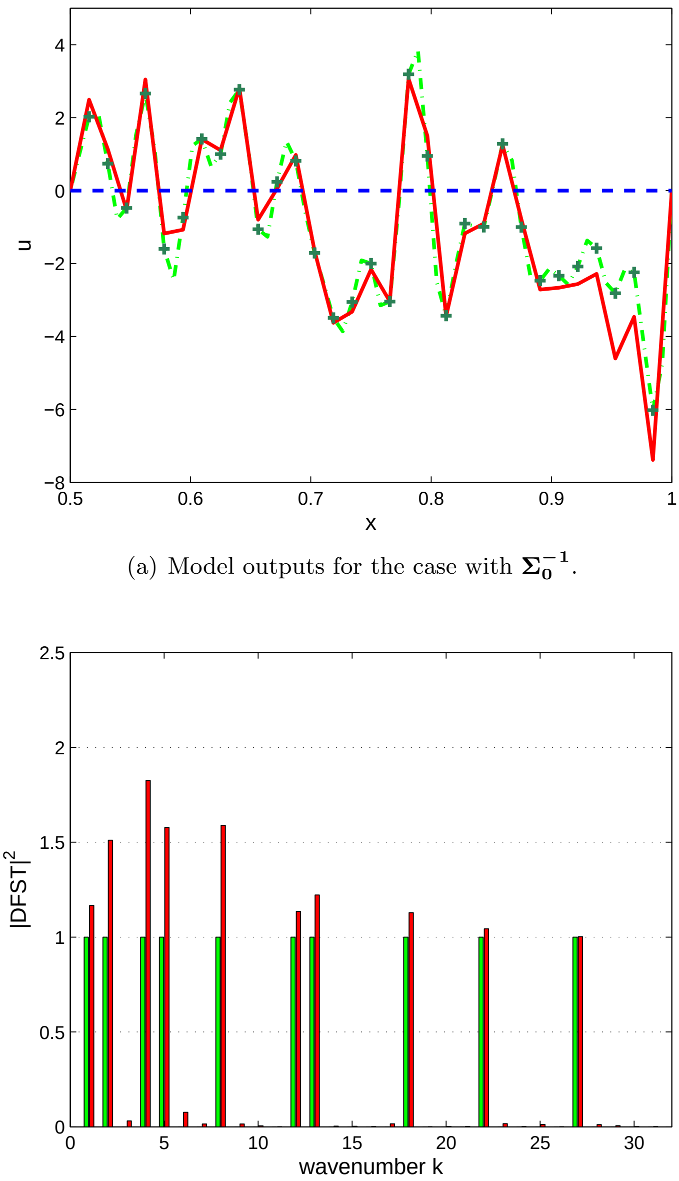
(no background term for the small scales and a small variance on the large scales), M is the number of LAM gridpoints.

In all four experiments the only thing to change is the matrix $\boldsymbol{\Sigma}^{\mathbf{-1}}$, everything else is kept the same. The reference trajectory at time $t = 0$ is a linear combination of sine waves with coefficient one and wavenumbers $\kappa = 1, 2, 4, 5, 8, 12, 13, 18, 22, 27$. As discussed in Section 7.4, the reference trajectory is used to provide the truth and the observations. The background is $\boldsymbol{x^b} = 0$ and therefore the background trajectory (as defined in Chapter 8) is also zero at all time levels. We have full, perfect observations and the inverse observational error covariance matrix is $\boldsymbol{R}^{\mathbf{-1}} = \frac{2}{M}\boldsymbol{I}$. As in Chapter 8, the LAM domain covers the right half of the parent domain, starting at the middle parent gridpoint.

Figures 9.3, 9.4, 9.5 and 9.6 show the model outputs and the power spectra for the case with $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}}$, $\boldsymbol{\Sigma}_{\mathbf{1}}^{\mathbf{-1}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{\mathbf{-1}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\beta}}^{\mathbf{-1}}$ respectively.

As can be seen in Figure 9.3(a), with $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}}$ the analysis is quite spiky and closely fits the truth. This is because for the case with $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}}$ there is no background term in the cost function. However, the background trajectory does still have some effect on the analysis because of the boundary conditions. As discussed in Section 8.1.2, this is why the analysis does not fit the truth exactly, despite having the resolution to capture all the wavenumbers present in the truth.

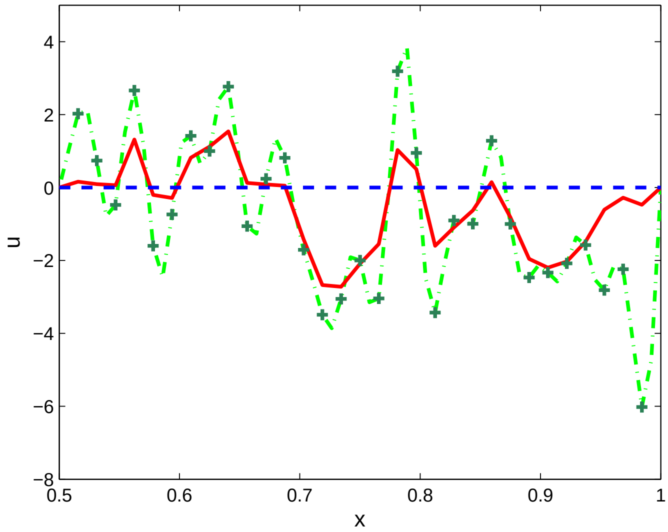
In contrast to the spiky analysis generated with $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}}$, for the case with $\boldsymbol{\Sigma}_{\mathbf{1}}^{\mathbf{-1}}$, which has a small variance on all wavenumbers, the analysis is much smoother, having been pulled back towards the zero background, as seen in Figure 9.4(a). The influence of the background can also be seen by considering the power spectra. In the power spectrum of the $\boldsymbol{\Sigma}^{\mathbf{-1}} = \mathbf{0}$ case, shown in Figure 9.3(b), the analysis has the same amplitude as the truth or greater. As described in Section 8.1.2, this increase in amplitude on the lower wavenumbers is caused by the inaccuracies at the boundaries between the background trajectory and the truth at times $t > 0$, as well as the increased diffusion in the LAM



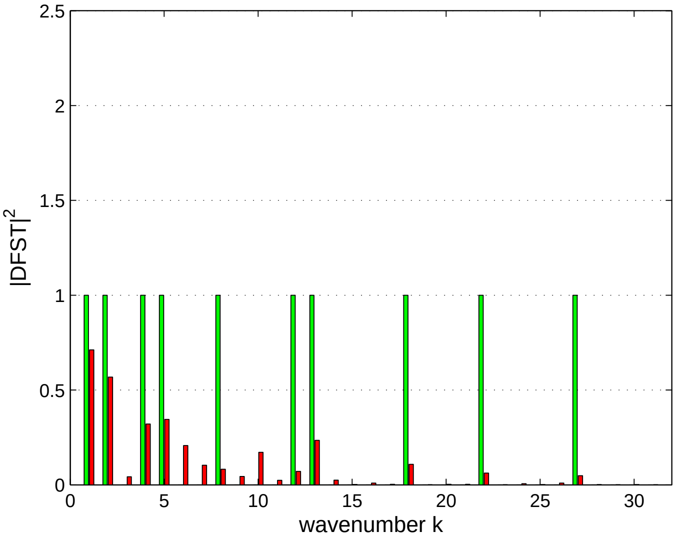
(a) Model outputs for the case with $\mathbf{\Sigma_0^{-1}}$.

(b) Power spectra for the case with $\mathbf{\Sigma_0^{-1}}$.

Figure 9.3: Model outputs and power spectra for the case with $\mathbf{\Sigma_0^{-1}}$. Green is the truth, red is the LAM analysis and blue is the background.

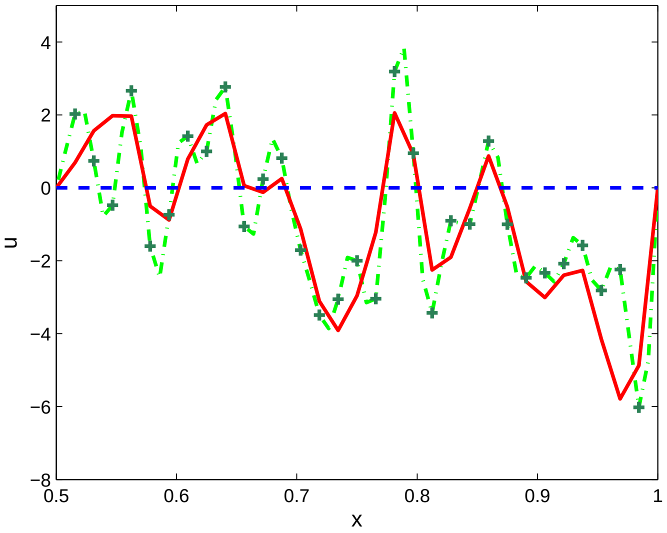


(a) Model outputs for the case with $\mathbf{\Sigma_1^{-1}}$.

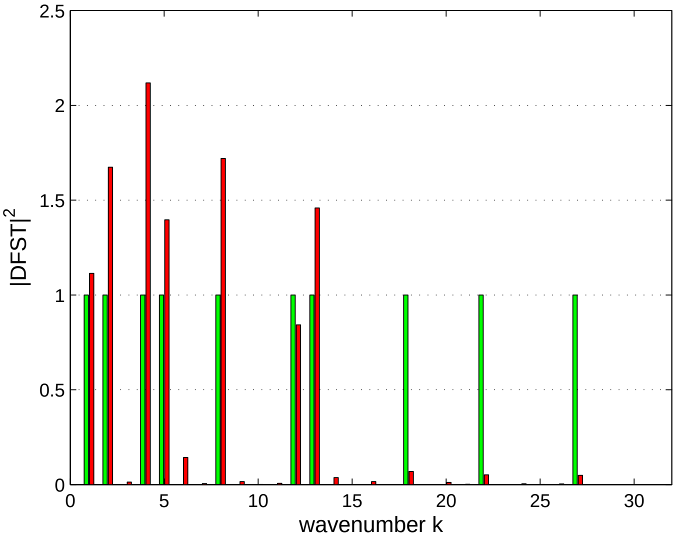


(b) Power spectra for the case with $\mathbf{\Sigma_1^{-1}}$.

Figure 9.4: Model outputs and power spectra for the case with $\mathbf{\Sigma_1^{-1}}$. Green is the truth, red is the LAM analysis and blue is the background.

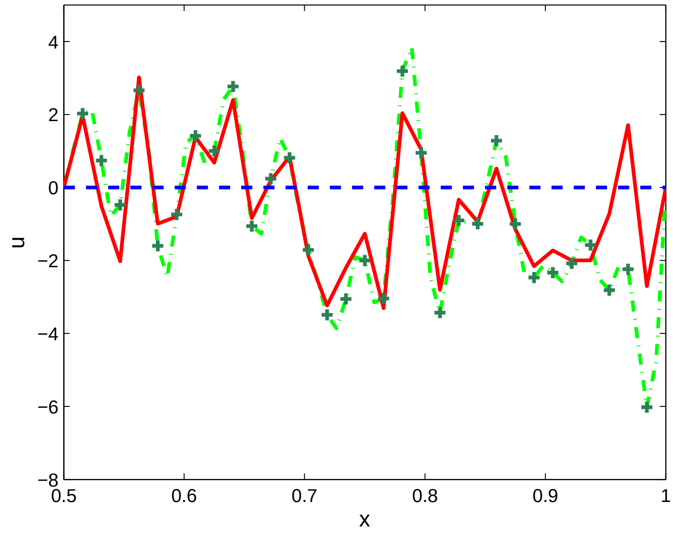


(a) Model outputs for the case with $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-\mathbf{1}}$.

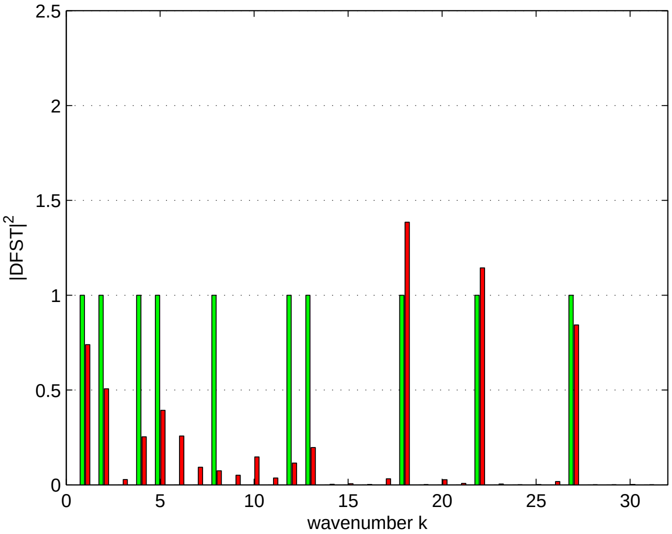


(b) Power spectra for the case with $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-\mathbf{1}}$.

Figure 9.5: Model outputs and power spectra for the case with $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-\mathbf{1}}$. Green is the truth, red is the LAM analysis and blue is the background.



(a) Model outputs for the case with $\mathbf{\Sigma}_{\beta}^{-1}$.



(b) Power spectra for the case with $\mathbf{\Sigma}_{\beta}^{-1}$.

Figure 9.6: Model outputs and power spectra for the case with $\mathbf{\Sigma}_{\beta}^{-1}$. Green is the truth, red is the LAM analysis and blue is the background.

caused by the difference in resolution compared to the truth. For the $\mathbf{\Sigma_0^{-1}}$ case, there is no background term in the cost function so the zero amplitude of the background trajectory is not affecting the power spectrum of the analysis, except through the LBCs. However, in Figure 9.4(b) we see that for the $\mathbf{\Sigma^{-1}}_1$ case the amplitude in the analysis is much lower than that of the truth, having been influenced by the zero background as well as the observations.

For the cases with split variances, $\mathbf{\Sigma_{\alpha}^{-1}}$ and $\mathbf{\Sigma_{\beta}^{-1}}$, we would hypothesize that the power spectra should resemble half of the power spectrum for the $\mathbf{\Sigma_0^{-1}}$ case and half of the one for the $\mathbf{\Sigma_1^{-1}}$ case, depending which way round we order the zeros and the ones. Looking at Figures 9.5(b) and 9.6(b), we see that this is indeed the case. Taking Figure 9.5(b) first, we see that for wavenumbers $1, \ldots, 15$, where $\alpha_i = 0$, the amplitude in the power spectrum agrees with that for the $\mathbf{\Sigma_0^{-1}}$ case. For wavenumbers $16, \ldots, 31$, where $\alpha_i = 1$, the amplitude in the power spectrum agrees with that for the $\mathbf{\Sigma_1^{-1}}$ case. The reverse can be seen for Figure 9.6(b).

If we now consider the model outputs in physical space, we see that these characteristics can still be observed. For the case with $\mathbf{\Sigma_{\alpha}^{-1}}$, shown in Figure 9.5(a), the analysis is smoother and for the case with $\mathbf{\Sigma_{\beta}^{-1}}$, shown in Figure 9.6(a), the analysis is more spiky. This is because in $\mathbf{\Sigma_{\alpha}^{-1}}$ we let the lower wavenumbers be influenced by the observations while the higher wavenumbers are dominated by the zero background. This results in little small scale detail and a smoother analysis. In contrast, in $\mathbf{\Sigma_{\beta}^{-1}}$ we let the higher wavenumbers be influenced by the observations while the lower wavenumbers are dominated by the zero background. This results in a lot of information in the small scales and a spikier analysis.

These results confirm the hypothesis that the wavenumbers can be constrained individually by the choice of matrix $\mathbf{\Sigma}$. Now that we have shown it is possible to constrain the scales seperately, we go on to investigate whether we can specify the matrix $\mathbf{\Sigma}$ in such a way as to improve the accuracy of the low wavenumbers whilst also accurately capturing the small scales.

9.3 More realistic examples with a non-zero background error covariance matrix $\boldsymbol{\Sigma}$

Having demonstrated in Section 9.2 the potential to use the matrix $\boldsymbol{\Sigma}$ to influence different wavenumbers we now want to try varying matrix $\boldsymbol{\Sigma}$ in more realistic cases.

We take our reference trajectory to be

$$u^r(x_j^r) = 5\sin(\pi x_j^r) + \sin(2\pi x_j^r) + \sin(36\pi x_j^r).$$

This combines one long-wave and two waves which are contained in the possible spectrum of the LAM, one low wavenumber and one high.

This reference trajectory is used to provide the observations. However, whereas in Chapter 8 we used perfect observations, we now add random noise to the observations. This random noise has variance $\sigma_o^2 = 0.25$ on all wavenumbers and the observation error covariance matrix is $\boldsymbol{R} = \sigma_o^2 \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix. We use the value $\sigma_o^2 = 0.25$ for our observation error variance because this is a typical error value for a thermometer, and in our advection-diffusion model the model variable is temperature. We have observations at every LAM gridpoint and timestep.

As in Chapter 8, we take the parent model to have $N = 16$ gridpoints and there are 10 parent timesteps in the assimilation window. The LAM covers the right-hand side of the parent grid, starting at the middle gridpoint. The LAM has four times the spatial resolution of the parent model and there are 16 LAM timesteps for every parent timestep. The reference trajectory has twice the spatial resolution of the LAM and four times the temporal resolution. The diffusion constant is still $\sigma = 0.001$ but now the advection speed is $c = 1$.

9.3.1 A background $\boldsymbol{x^b}$ with no random noise

Given the presumption that longer waves are accurately represented in the model providing the background, we choose a background that correctly contains these waves but does not contain any higher resolution information. The background at $t = 0$ is

$$u^p(x_j^r) = 5\sin(\pi x_j^p) + \sin(2\pi x_j^p).$$

The background trajectory is then generated by first running the parent model forward from these initial conditions. The background is then interpolated to the LAM grid and re-run at the LAM resolution using boundary conditions provided from the parent resolution run. Generating the background trajectory in this way agrees with a set-up where the background is provided by a parent model with a coarser resolution but larger domain size than the LAM.

Varying the choice of matrix $\mathbf{\Sigma}$, we run the data assimilation (spectral scheme) with the same observations, background and background trajectory, to compare how the choice of matrix $\mathbf{\Sigma}$ affects the different scales in the analysis.

We use $\mathbf{\Sigma_0^{-1}} = 0$ (no background term in the cost function), $\mathbf{\Sigma}_A = 0.05^2\boldsymbol{I}$ (the same small variance on all wavenumbers), $\mathbf{\Sigma}_\alpha = \text{diag}\{\alpha_i^2\}$ and $\mathbf{\Sigma}_\beta = \text{diag}\{\beta_i^2\}$ where

$$\alpha_i = \begin{cases} 0.005 & \text{if } i = 1, \ldots, 15 \\ 0.5 & \text{if } i = 16, \ldots, M \ , \end{cases}$$

and

$$\beta_i = \begin{cases} 0.005 & \text{if } i = 1 \\ 0.05 & \text{if } i = 2, \ldots, 15 \\ 0.5 & \text{if } i = 16, \ldots, M \ . \end{cases}$$

$\mathbf{\Sigma}_\beta$ has a very small variance on wavenumber $k = 1$, a small variance on the other lower wavenumbers and a variance equal to that on the observations on the higher wavenumbers. $\mathbf{\Sigma}_\alpha$ has the same variance as $\mathbf{\Sigma}_\beta$ on the higher wavenumbers but has an extremely small variance on all the lower wavenumbers, instead of just $k = 1$.

We choose $\mathbf{\Sigma}_\alpha$ and $\mathbf{\Sigma}_\beta$ like this because, as discussed in Section 8.1.2, the LAM over estimates the low wavenumbers due to the LBCs. The low wavenumbers also contain the aliased long-wave information, as shown in Section 8.2. By placing a smaller variance on the low wavenumbers we aim to constrain these scales to match the background, which is assumed to accurately capture the large scales. The higher wavenumbers are given a larger variance with the aim of letting the observations have a greater influence on the small scales.

Figure 9.7 shows the model outputs for the four choices of matrix $\mathbf{\Sigma}$. The four analyses are plotted offset from each other on the same axis to enable them to be compared more easily. The reference trajectory (truth) is shown in green, the background in blue

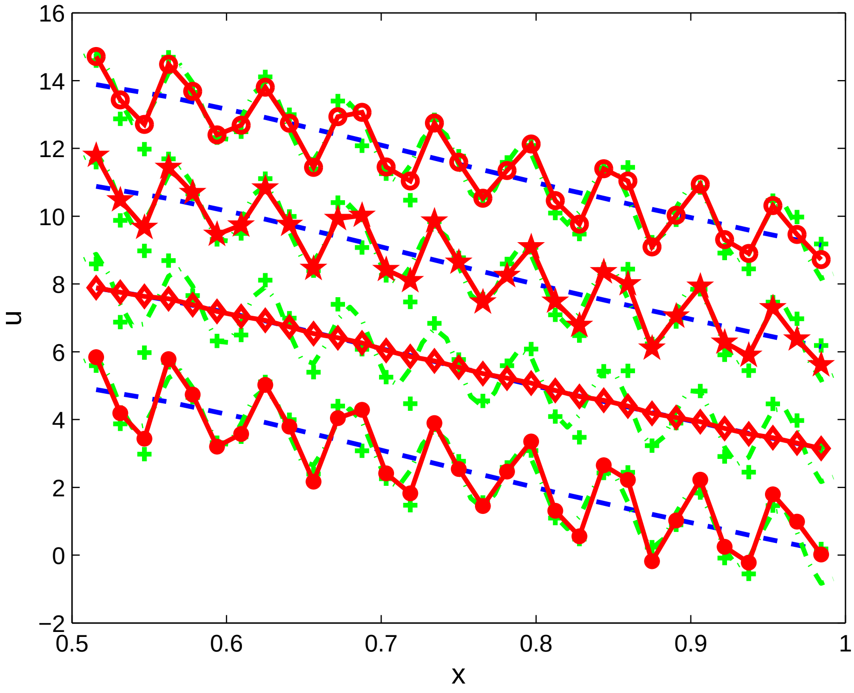


Figure 9.7: Model outputs at $t = 0$ for $\mathbf{\Sigma_0^{-1}}$, $\mathbf{\Sigma}_A$, $\mathbf{\Sigma}_\alpha$ and $\mathbf{\Sigma}_\beta$. Green is the reference trajectory (truth), blue is the background and red is the analysis. The green crosses are observations.

and the analyses are in red. The observations are shown by green crosses. The bottom analysis is generated using $\mathbf{\Sigma^{-1}}_0$, the analysis generated using $\mathbf{\Sigma}_A$ is then offset by three, the analysis generated using $\mathbf{\Sigma}_\alpha$ is then offset by six and the analysis generated using $\mathbf{\Sigma}_\beta$ is offset by nine.

As we would expect, the analysis generated with $\mathbf{\Sigma_0^{-1}}$ closely fits the observations due to there being no background term in the cost function. In comparison, the analysis generated with $\mathbf{\Sigma}_A$, which has an extremely small variance on all wavenumbers, closely fits the background and has little high resolution detail. This is because the variance on the background is smaller than the variance on the observations by a factor of a hundred. This results in the analysis being strongly constrained by the background and allows the observations to have little effect. The benefit of the high resolution detail in the observations is therefore not felt by the analysis. The analyses generated with $\mathbf{\Sigma} = \alpha_i^2 \boldsymbol{I}$ and $\mathbf{\Sigma} = \beta_i^2 \boldsymbol{I}$ fit the observations closely but have slightly lower amplitude than the $\mathbf{\Sigma_0^{-1}}$ case, due to the influence of the background, though this is difficult to observe by eye.

The differences in the analyses can be better understood by considering the power

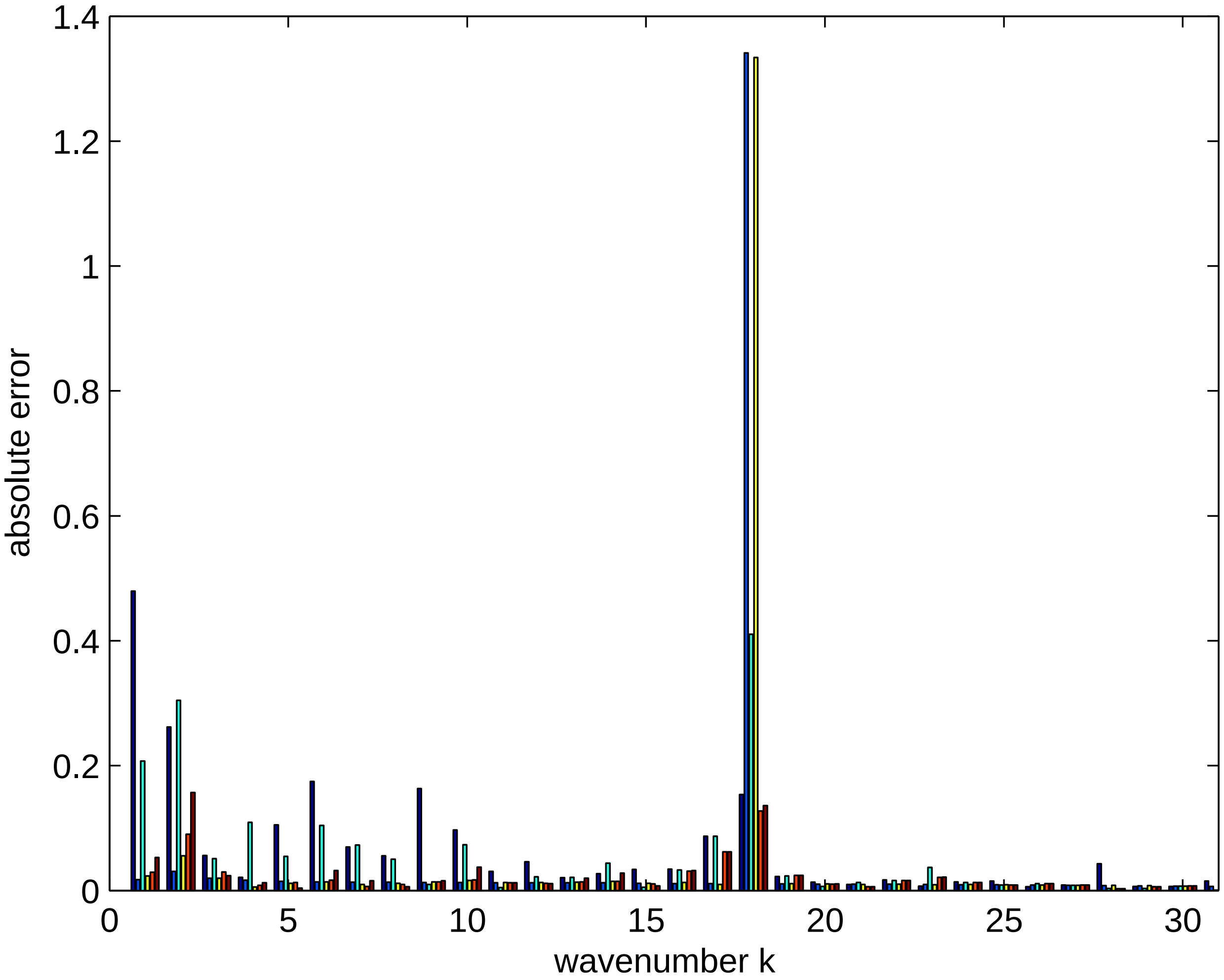


Figure 9.8: Errors in the power spectrum at $t = 0$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma_0^{-1}}$ (light blue), $\mathbf{\Sigma}_A$ (yellow), $\mathbf{\Sigma}_\alpha$ (orange) and $\mathbf{\Sigma}_\beta$ (red).

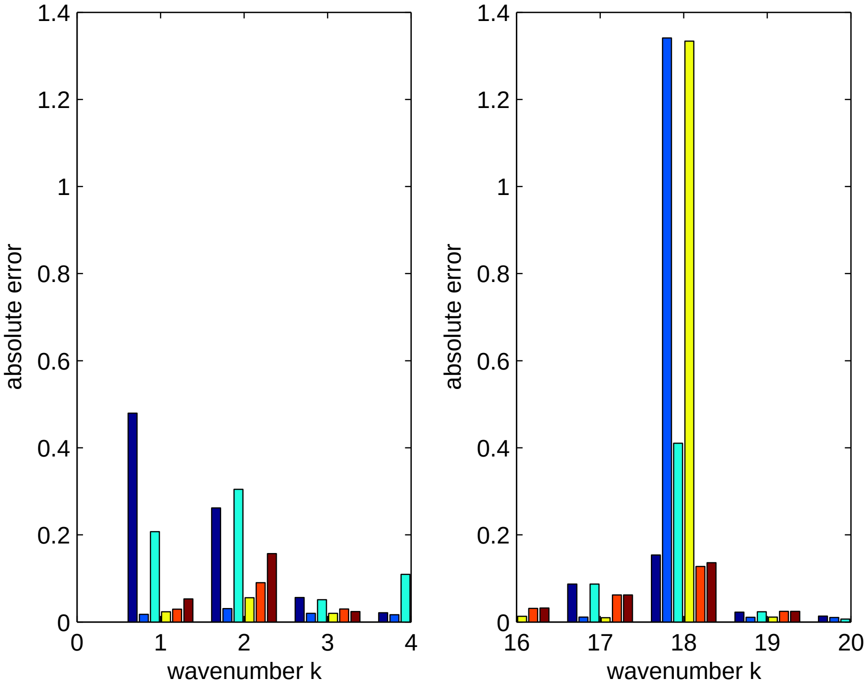


Figure 9.9: Errors in the power spectrum at $t = 0$, a close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma_0^{-1}}$ (light blue), $\mathbf{\Sigma}_A$ (yellow), $\mathbf{\Sigma}_\alpha$ (orange) and $\mathbf{\Sigma}_\beta$ (red). A close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$.

spectra. Although the power spectra of all four analyses are similar there are subtle differences that reveal properties of the analyses. To make the differences in the power spectra more apparent, instead of plotting the power spectra themselves we instead plot the absolute error in the power spectra. These absolute errors are plotted in Figure 9.8. The absolute errors are the magnitude of the difference between the power spectra of the analyses and that of the reference trajectory (truth). Also shown in Figure 9.8 are the absolute errors in the observations (dark blue) and the background (mid blue). As there are a lot of wavenumbers the columns are quite tightly packed and difficult to see. Therefore, a close up of the wavenumbers contained in the reference trajectory is shown in Figure 9.9. Light blue is the error in the $\mathbf{\Sigma_0^{-1}}$ analysis, yellow the $\mathbf{\Sigma}_A$ analysis, orange the $\mathbf{\Sigma}_\alpha$ analysis and red the $\mathbf{\Sigma}_\beta$ analysis.

Concentrating on the errors at $k = 1$ and $k = 18$ to begin with, we see in Figure 9.9 that the observations (dark blue) have errors at both wavenumbers (≈ 0.5 at $k = 1$ and ≈ 0.1 at $k = 18$). This is as we would expect having added random noise to the observations. The background has a small error (< 0.1) at $k = 1$, caused by the difference in resolution compared to the reference trajectory. It also has a large error (> 1.2) at $k = 18$, caused by the lack of high resolution information in the background. We plot these errors, as well as those of the analyses, because we can use these errors to understand the errors in the analyses.

We consider the $\mathbf{\Sigma_0^{-1}}$ case first. Having the matrix $\mathbf{\Sigma^{-1}}$ being equal to zero means that there is no background term in the cost function, leaving just the observation term. Despite this, the power spectrum of the $\mathbf{\Sigma_0^{-1}}$ case (light blue) does not exactly match that of the observations. This is because despite the lack of background term in the cost function, the background trajectory does still influence the analysis through the lateral boundary conditions and the buffer zone, as discussed in Section 8.1.2. This influence could be what causes the analysis to be more accurate than the observations at $k = 1$. However, it is also worth noting that at $k = 18$, as well as many other wavenumbers, the error in the $\mathbf{\Sigma_0^{-1}}$ analysis is greater than that of the observations. This is also due to the influence of the background trajectory through the LBCs.

Now considering the error in the power spectrum of the analysis generated using $\mathbf{\Sigma}_A$ (yellow), which has an extremely small variance on all wavenumbers, we see that it has a very small error at $k = 1$. This is as we would expect, due to the analysis being constrained to closely match the background. At $k = 18$ however, there is a large error, again caused by the constraining of the analysis to match the background via our choice of $\mathbf{\Sigma}$. By having a much smaller variance on the background compared to the observations, the analysis is heavily constrained to match the background and the information contained in the observations about wavenumber $k = 18$ cannot influence the analysis.

Using these patterns in the power spectrum we can understand better the analyses for the $\mathbf{\Sigma}_\alpha$ and $\mathbf{\Sigma}_\beta$ cases. In each analysis there are only small errors (< 0.2) at both $k = 1$ and $k = 18$, as well as at the other wavenumbers. This demonstrates that by varying the values in $\mathbf{\Sigma}$ we can constrain the scales differently. By placing a small variance on the lower wavenumbers they have benefited from the accurate large scale information in the background, while a larger variance on the higher wavenumbers have allowed those wavenumbers to be influenced by the high resolution information in the observations.

It is also interesting to note the difference in the error at $k = 2$ between the $\mathbf{\Sigma}_\alpha$ and $\mathbf{\Sigma}_\beta$ cases. Due to the random error, the observations have an error (> 0.2) at $k = 2$.

In both the $\mathbf{\Sigma}_\alpha$ and $\mathbf{\Sigma}_\beta$ cases the influence of the background has meant the errors in the analyses are less than those of the observations. However, the error at $k = 2$ in the $\mathbf{\Sigma}_\beta$ case is greater than that in the $\mathbf{\Sigma}_\alpha$ case. This is caused by the differences in the variances placed on the low wavenumbers. While both cases have the same, extremely small, variance on $k = 1$, $\mathbf{\Sigma}_\beta$ has a larger variance on $k = 2, \ldots, 15$, allowing the observations to have more influence. In contrast, $\mathbf{\Sigma}_\alpha$ has the same, extremely small, variance for $k = 2, \ldots, 15$ as it has for $k = 1$. This constrains more low wavenumbers to match the background and reduces the error in the analysis. This demonstrates that as well as constraining wavenumber $k = 1$, there is also a benefit in constraining more of the low wavenumbers.

These results demonstrate that we can choose $\mathbf{\Sigma}$ in such a way as to influence the large and small scales differently. Also by constraining the low wavenumbers to match the background we can better represent the long-wave information, which is aliased to low wavenumbers on the LAM domain as discussed in Chapter 8, and reduce the errors resulting from the LBCs causing the LAM analysis to overestimate the low wavenumbers, as discussed in Section 8.1.2.

In reality however, the parent model would never capture the large scales perfectly (although the zero boundary conditions assume that we do). Therefore, we now repeat our experiments using a background that has random noise added to the large scales.

9.3.2 A background with random noise

We again choose a background that correctly contains the longer waves but does not contain any higher resolution information, following the presumption that longer waves are accurately represented in the model providing the background. However, the parent model output will never be perfect, therefore, we now add some random noise to this background. The background is generated by first running the parent model forward from the initial conditions

$$u^p(x_j^r) = 5\sin(\pi x_j^p) + \sin(2\pi x_j^p).$$

This is then interpolated to the LAM grid as before. However, random noise is now added to the initial conditions, on the LAM resolution, at the wavenumbers contained in the spectrum of the parent model. These new initial conditions give us the background for the LAM. The background trajectory is then generated by running the background forward, at the LAM resolution, using the LBCs from the parent model run. The random noise has variance 0.5^2 and is added to wavenumbers $k = 1, \ldots, 7$. This variance is equal to that of the noise added to the observations. All the other parameters are kept equal to those in Section 9.3.1.

Varying the choice of $\boldsymbol{\Sigma}$, we again run the data assimilation (spectral scheme) with the same observations and background, as we did in Section 9.3.1. We compare how the choice of matrix $\boldsymbol{\Sigma}$ affects how the different scales are treated in the analysis and whether we can specify the matrix $\boldsymbol{\Sigma}$ in such a way to improve the low wavenumbers in a LAM analysis. These low wavenumbers are adversely affected by the LBCs as well as being where the long-wave information is aliased onto in the LAM analysis. We therefore want to capture these low wavenumbers as accurately as possible.

The same variance on all wavenumbers compared to splitting the variances

We begin by using $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}} = 0$, $\boldsymbol{\Sigma}_A = 0.05^2\boldsymbol{I}$ (the same, small variance on all wavenumbers), $\boldsymbol{\Sigma}_B = 0.5^2\boldsymbol{I}$ (a variance on all wavenumbers matching that of the random noise on wavenumbers $k = 1, \ldots, 7$ of the background) and $\boldsymbol{\Sigma}_\gamma = \mathrm{diag}\{\gamma_i^2\}$, where

$$\gamma_i = \begin{cases} 0.5 & \text{if } i = 1, \ldots, 7 \\ 5.0 & \text{if } i = 8, \ldots, M \ , \end{cases}$$

(the correct variance on wavenumbers $k = 1, \ldots, 7$ and a large variance elsewhere to allow greater influence from the observations).

Figure 9.10 shows the model outputs for the four choices of $\boldsymbol{\Sigma}$. The four analyses are plotted offset from each other on the same axis to enable them to be compared more easily. The reference trajectory (truth) is shown in green, the background in blue and the analyses are in red. The observations are shown by green crosses. The bottom analysis (marked by red dots) is generated using $\boldsymbol{\Sigma}_{\mathbf{0}}^{\mathbf{-1}}$, the analysis generated using $\boldsymbol{\Sigma}_A$ (marked by red diamonds) is then offset by three, the analysis generated using $\boldsymbol{\Sigma}_B$ (marked by red stars) is then offset by six and the analysis generated using $\boldsymbol{\Sigma}_\gamma$ (marked by red cirles) is offset by nine.

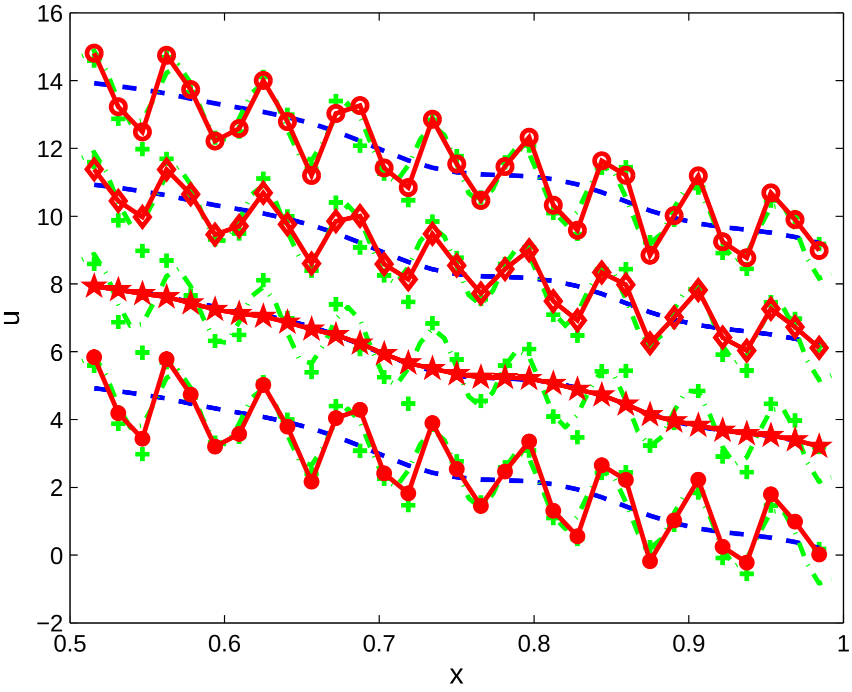


Figure 9.10: Model outputs at $t = 0$ for $\mathbf{\Sigma_0^{-1}}$, $\mathbf{\Sigma}_A$, $\mathbf{\Sigma}_B$ and $\mathbf{\Sigma}_\gamma$. Green is the reference trajectory (truth), blue is the background and the green crosses are observations. The analyses are shown in red. The bottom analysis (marked by red dots) is generated using $\mathbf{\Sigma_0^{-1}}$, the analysis generated using $\mathbf{\Sigma}_A$ (marked by red diamonds) is then offset by three, the analysis generated using $\mathbf{\Sigma}_B$ (marked by red stars) is then offset by six and the analysis generated using $\mathbf{\Sigma}_\gamma$ (marked by red cirles) is offset by nine.

As we would expect, the analysis generated with $\mathbf{\Sigma_0^{-1}}$ again closely fits the observations due to having no background term in the cost function. In comparison, the analysis generated with $\mathbf{\Sigma}_A$, which has an extremely small variance on all wavenumbers, closely fits the background and has little amplitude in the high resolution oscillations. This is because of the extremely low background variance, which constrains the analysis to closely match the background. The other two analyses both appear to be very similar to that of the $\mathbf{\Sigma_0^{-1}}$ case, and while there are subtle differences between them these are difficult to observe by eye. To better understand the differences between the analyses we consider again the absolute errors in the power spectra.

Figure 9.11 shows the absolute errors in the power spectra. Again, as there are a lot of wavenumbers the columns are quite tightly packed and difficult to see. Therefore, a close up of the wavenumbers contained in the reference trajectory is shown in Figure 9.12. Dark blue is the error in the observations, mid blue is the error in the background, light blue is the error in the $\mathbf{\Sigma_0^{-1}}$ analysis, yellow the $\mathbf{\Sigma}_A$ analysis, orange the $\mathbf{\Sigma}_B$ analysis and red the $\mathbf{\Sigma}_\gamma$ analysis.

Concentrating on the errors at $k = 1$ and $k = 18$ to begin with, we see in Figure 9.12 that the observations have errors at both $k = 1$ (≈ 0.5) and $k = 18$ (≈ 0.2). This is as we would expect having added random noise to the observations. The background also has an error (≈ 0.15) at $k = 1$, caused by the random noise. It also has a large error (> 1.2) at $k = 18$, caused by the lack of high resolution information in the background. We plot these errors, as well as those of the analyses, so that we can use these errors to understand the errors in the analyses.

If we consider the $\mathbf{\Sigma^{-1}}_0$ case first, we see that although there is no background term in the cost function, the power spectrum of the analysis does not exactly match that of the observations. In fact, the error at $k = 18$ is almost double that of the observations. This increase in error is caused because although there is no background term in the cost function, the background does still influence the analysis through the lateral boundary conditions and the buffer zone, as discussed in Section 8.1.2.

Now considering the analysis generated with $\mathbf{\Sigma}_A$, which has an extremely small variance on all wavenumbers, we see that it has a small error at $k = 1$. This is as we would expect, due to it being constrained to closely match the background. In fact, the error at $k = 1$

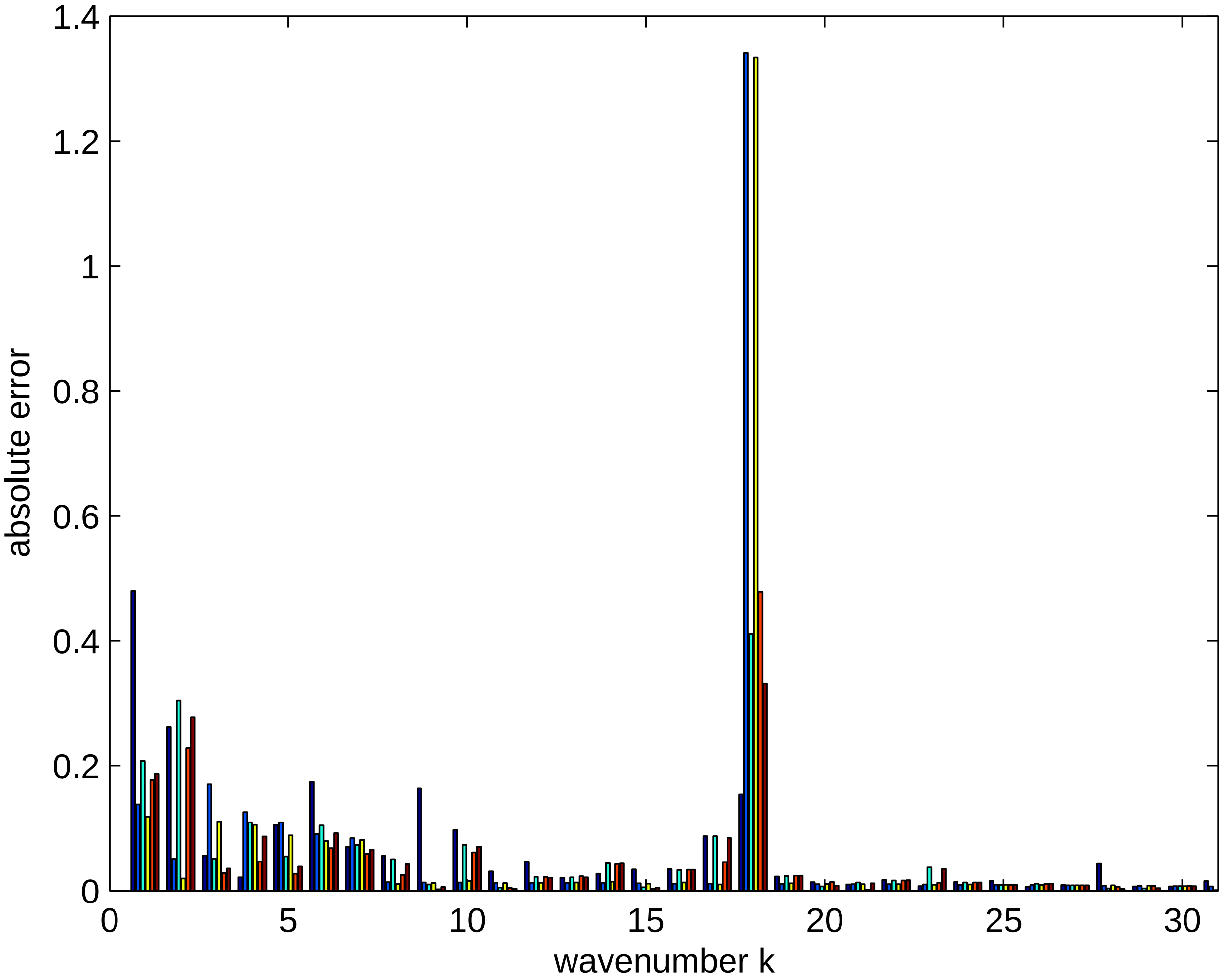


Figure 9.11: Errors in the power spectrum at $t = 0$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma_0^{-1}}$ (light blue), $\mathbf{\Sigma}_A$ (yellow), $\mathbf{\Sigma}_B$ (orange) and $\mathbf{\Sigma}_\gamma$ (red).

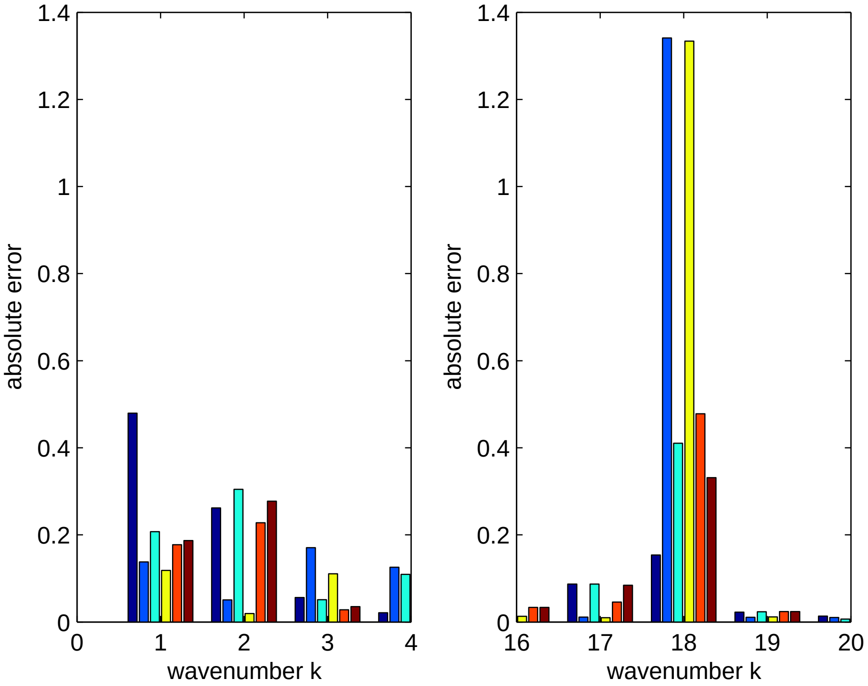


Figure 9.12: Errors in the power spectrum at $t = 0$, a close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma_0^{-1}}$ (light blue), $\mathbf{\Sigma}_A$ (yellow), $\mathbf{\Sigma}_B$ (orange) and $\mathbf{\Sigma}_\gamma$ (red). A close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$.

is actually slightly lower than that of the background. This demonstrates the possibility of improving the representation of low wavenumbers on the LAM, even compared to the background, by over constraining the low wavenumbers in $\mathbf{\Sigma}$. At $k = 18$ however, there is a large error (> 1.2), again caused by the constraining of the analysis to match the background via our choice of $\mathbf{\Sigma}$.

The $\mathbf{\Sigma}_B$ case has the same variance on all wavenumbers and this is the correct variance from the added random noise on wavenumbers $k = 1, \ldots, 7$. It also means we have the same variance on the background as on the observations. This analysis is fairly accurate at all wavenumbers. It has a small error at $k = 1$ but it is worse than the $\mathbf{\Sigma}_A$ case. This demonstrates the benefit in overconstraining the low wavenumbers, as $\mathbf{\Sigma}_A$ has a lower variance than $\mathbf{\Sigma}_B$. Using the correct variances does not necessarily give the best results. The error at $k = 18$ is also worse than some of the other analyses. This is because the variance on the background is still too small at high wavenumbers.

In the $\mathbf{\Sigma}_\gamma$ case we keep the correct variance for the lower wavenumbers, but increase the variance on the higher wavenumbers. This agrees with the presumption that while the larger scales are accurately represented in the background, the small scales are not. Comparing this analysis with the $\mathbf{\Sigma}_B$ case, which has a smaller variance than $\mathbf{\Sigma}_\gamma$ on the higher wavenumbers, we see that while it has roughly the same error at $k = 1$ it has generally worse errors at most other wavenumbers. This demonstrates the need for balance. While the larger background variances at higher wavenumbers in $\mathbf{\Sigma}_\gamma$ have allowed the observations to influence $k = 18$, making this noticably better in the $\mathbf{\Sigma}_\gamma$ case compared to the $\mathbf{\Sigma}_B$ case, the larger variances have also allowed the errors in the observations to negatively influence the analysis at other wavenumbers. This highlights that while the background does not accurately represent the small scales, it is still important to not use overly large variances in $\mathbf{\Sigma}$.

By overconstraining the low wavenumbers, the $\mathbf{\Sigma}_A$ analysis is more accurate than the background at low wavenumbers. By splitting the scales in $\mathbf{\Sigma}_\gamma$ we can constrain the low wavenumbers to match the background while allowing the observations to influence the higher wavenumbers. These results show the possibility of specifying $\mathbf{\Sigma}$ in such a way as to enable the LAM analysis to accurately capture all the scales present in the truth. Therefore, we now try more choices of $\mathbf{\Sigma}$, altering the variances on the different wavenumbers.

Comparing choices of $\mathbf{\Sigma}$ with different variances on different wavenumbers

We aim to choose the variances in the matrix $\mathbf{\Sigma}$ in such a way as to reduce the error in the LAM analysis, particularly at low wavenumbers. From Section 8.1.2 we know that the LBCs cause the LAM to over estimate the low wavenumbers. From Section 8.2 we know that the long-wave information is also aliased to the low wavenumbers. By placing a smaller variance on the low wavenumbers we aim to constrain these scales to match the background, which is assumed to accurately capture the large scales. The higher wavenumbers are given a larger variance with the aim of letting the observations have a greater influence on the smaller scales. We now compare different choices of matrix $\mathbf{\Sigma}$. All have variances split between the different scales, but the actual variances used at the different scales varies between them.

We have $\mathbf{\Sigma}_\mu = \mathrm{diag}\{\mu_i^2\}$, $\mathbf{\Sigma}_\nu = \mathrm{diag}\{\nu_i^2\}$, $\mathbf{\Sigma}_\omega = \mathrm{diag}\{\omega_i^2\}$ and $\mathbf{\Sigma}_\eta = \mathrm{diag}\{\eta_i^2\}$, where

$$\mu_i = \begin{cases} 0.05 & \text{if } i = 1, \dots, 7 \\ 0.5 & \text{if } i = 8, \dots, M \ , \end{cases}$$

$$\nu_i = \begin{cases} 0.05 & \text{if } i = 1, \dots, 7 \\ 5.0 & \text{if } i = 8, \dots, M \ , \end{cases}$$

$$\omega_i = \begin{cases} 0.005 & \text{if } i = 1, \dots, 7 \\ 0.5 & \text{if } i = 8, \dots, M \ , \end{cases}$$

$$\eta_i = \begin{cases} 0.005 & \text{if } i = 1, 2 \\ 0.05 & \text{if } i = 3, \dots, 7 \\ 0.5 & \text{if } i = 8, \dots, M \ . \end{cases}$$

As we saw in Figure 9.10, the subtleties between the different analyses are difficult to observe in physical space. We therefore do not plot the model outputs in physical space and just consider the absolute errors in the power spectra.

Figure 9.13 shows the absolute errors in the power spectra. As before, since the columns are quite tightly packed and difficult to see, a close up of the wavenumbers contained in the reference trajectory (truth) is shown in Figure 9.14. Dark blue is the error in the observations, mid blue is the error in the background, light blue is the error in the $\mathbf{\Sigma}_\mu$ analysis, yellow the $\mathbf{\Sigma}_\nu$ analysis, orange the $\mathbf{\Sigma}_\omega$ analysis and red the $\mathbf{\Sigma}_\eta$ analysis.

If we concentrate on $k = 1$ to start with, we see in Figures 9.13 and 9.14 that all four analyses are at least marginally better than the background. This demonstrates the benefit of over constraining the low wavenumbers, as in all four choices of $\mathbf{\Sigma}$ the variance on wavenumber $k = 1$ is lower than the true variance. This is further demonstrated by the smallest errors at $k = 1$ being for the analyses that most constrain wavenumber $k = 1$, namely $\mathbf{\Sigma}_\omega$ and $\mathbf{\Sigma}_\eta$.

It is also interesting that the smallest error at $k = 1$ is on the $\mathbf{\Sigma}_\eta$ case, even though it only constrains the first two wavenumbers by the highest amount. In contrast, the $\mathbf{\Sigma}_\omega$ case has strongly constrained all of the first seven wavenumbers, but this actually has a slightly detrimental effect on the errors. At all of the first seven wavenumbers, and several others, the $\mathbf{\Sigma}_\omega$ case has larger errors than the $\mathbf{\Sigma}_\eta$ case. This shows that while it

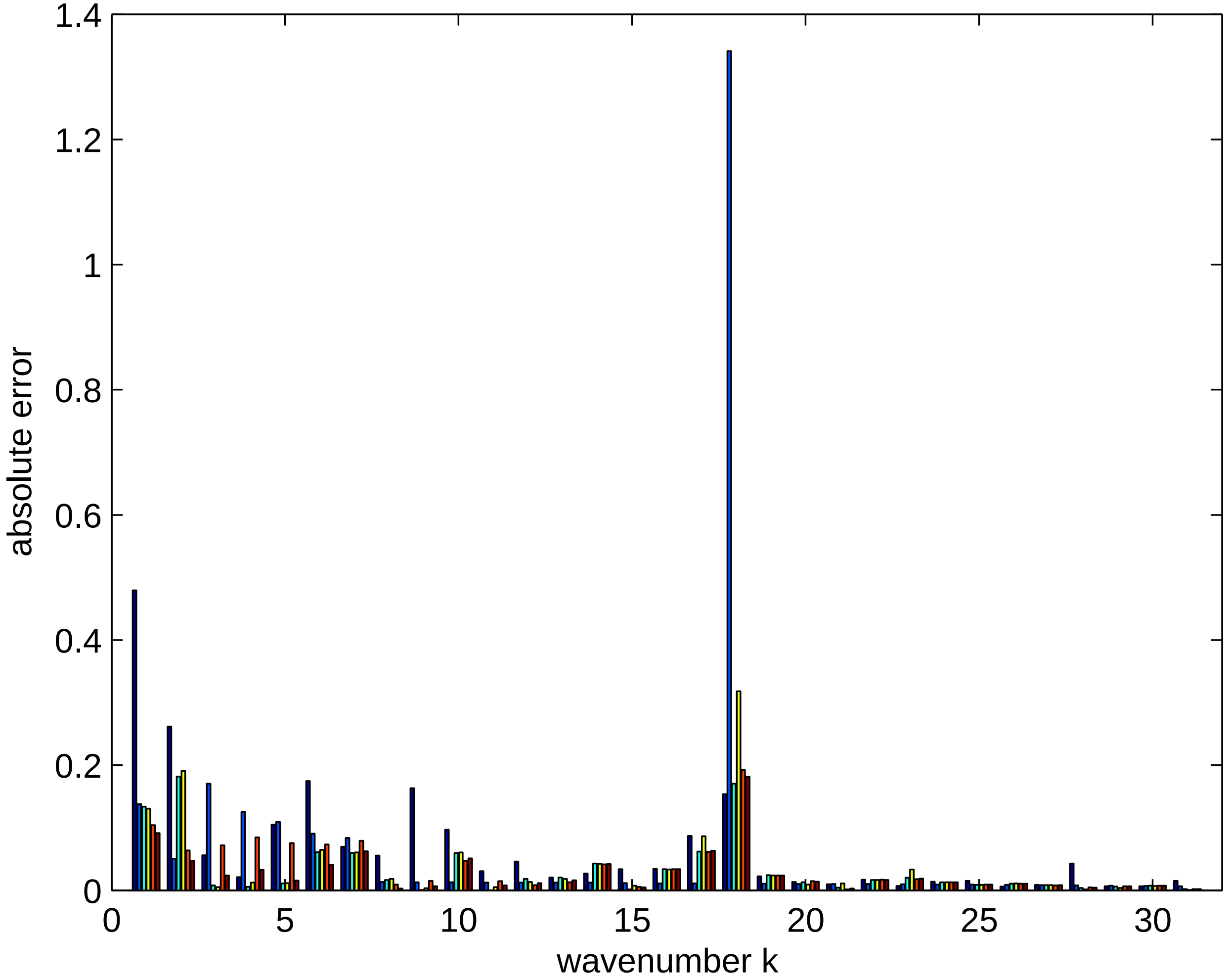


Figure 9.13: Errors in the power spectrum at $t = 0$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma}_\mu$ (light blue), $\mathbf{\Sigma}_\nu$ (yellow), $\mathbf{\Sigma}_\omega$ (orange) and $\mathbf{\Sigma}_\eta$ (red).

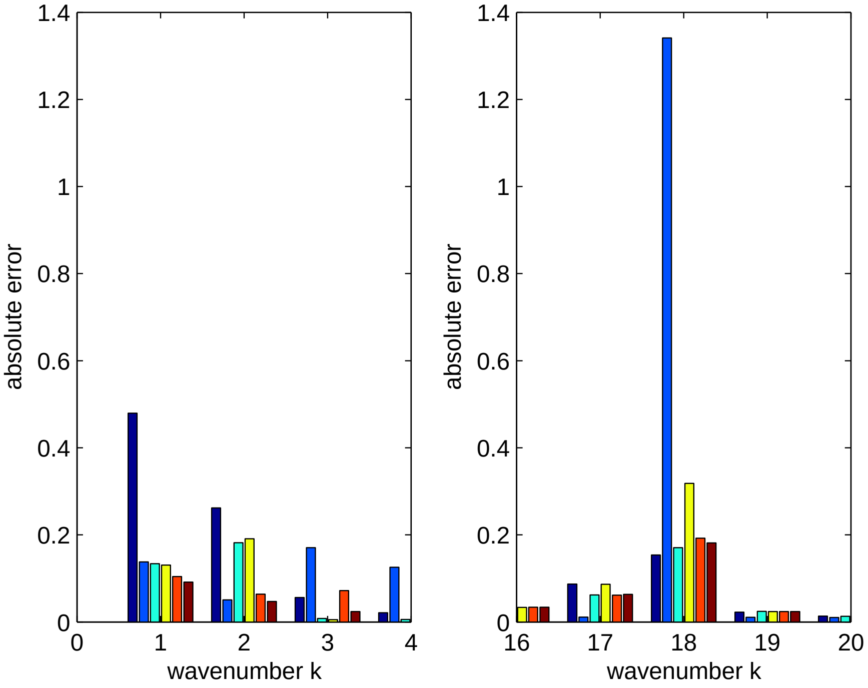


Figure 9.14: Errors in the power spectrum at $t = 0$, a close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$ for the observations (dark blue), the background (mid blue), the analyses generated with $\mathbf{\Sigma}_\mu$ (light blue), $\mathbf{\Sigma}_\nu$ (yellow), $\mathbf{\Sigma}_\omega$ (orange) and $\mathbf{\Sigma}_\eta$ (red). A close up of $k = 0, \ldots, 4$ and $k = 16, \ldots, 20$.

is beneficial to constrain the lowest wavenumbers, it is important to maintain a balance and not overly constrain too many wavenumbers.

The benefit of overconstraining the low wavenumbers is also evident by comparing the $\mathbf{\Sigma}_\gamma$ analysis (red) in Figure 9.12 and the $\mathbf{\Sigma}_\nu$ analysis (yellow) in Figure 9.14. $\mathbf{\Sigma}_\gamma$ and $\mathbf{\Sigma}_\nu$ have the same variances on the higher wavenumbers but while $\mathbf{\Sigma}_\gamma$ has the correct variance on the low wavenumbers (as specified by the added random noise), $\mathbf{\Sigma}_\nu$ has an overly small variance on these wavenumbers. Comparing the errors in the power spectrum, we see that while both have similar errors at $k = 18$, at $k = 1$ the $\mathbf{\Sigma}_\nu$ analysis is more accurate. This improvement in accuracy is due to overconstraining the low wavenumbers.

Now considering $k = 18$, we see from Figure 9.13 and 9.14 that the largest errror is in the $\mathbf{\Sigma}_\nu$ case. This choice of $\mathbf{\Sigma}$ has a variance of 5 on all wavenumbers above $k = 7$, the largest variance of any of the analyses. This demonstrates an important point. While the small scales are not well represented in the background and it is necessary to set a large enough variance in $\mathbf{\Sigma}$ such as to allow the observations to influence these wavenumbers, it is also important to maintain a balance between the background and observation variances and not set the variances in $\mathbf{\Sigma}$ too large, even at high wavenumbers. The other three analyses all have the same variance at wavenumbers above $k = 7$ and their errors at these wavenumbers are all relatively similar. One may be slightly better at one location but another is better elsewhere.

These results demonstrate the potential of $\mathbf{\Sigma}$ to affect the different length scales in an analysis. By placing a smaller variance on the lower wavenumbers we can not only constrain the large scales so as to not degrade them in the LAM analysis, we can actually improve them compared to the background. At the same time we can accurately represent the small scales by balancing the variance on the higher wavenumbers in the background with that on the observations.

9.4 Comparisons with other work

As described in Section 4.2.1, there have been other attempts to improve the problems of accurately representing all the scales in a LAM analysis.

Ballard et al [2] does a complete split of the scales, merging the large scales from a parent model run with the small scales from the LAM. We could accomplish a similar distinct splitting of the scales by placing a small variance on the low wavenumbers to constrain those to the background, and a large variance on the high wavenumbers to allow those to be influenced by the observations. A version of scale splitting using our method was demonstrated in Section 9.2. Although both methods seem to perform a similar scale splitting, there are also distinct differences between splitting the scales using our method and that used by Ballard et al. In particular, in the Ballard et al method the LAM data assimilation is first run as normal using a background which is provided by a previous LAM forecast. The small scales from this analysis are then blended with the large scales from a parent analysis to generate the final LAM analysis, which is then used to generate the forecast. However, by generating the scales in separate analyses and then blending them, it is not guaranteed that the resulting analysis obeys the model equations. In our method however, the generation of the LAM analysis and the scale splitting is all done in one step. This guarantees an analysis which obeys the model equations and potentally creates a smoother transition between the large and small scales. However, it also means the small scales do not benefit from the information in previous LAM runs, as the background is provided by the parent model.

Guidard and Fischer [34] constrain the large scales to those of the model providing the boundary conditions by means of an extra term J_k in the cost function. Comparing to our method, constraining the large scales in the LAM analysis to those of the parent analysis providing the boundary conditions is precisely what we are doing when we set a small variance on the low wavenumbers in $\boldsymbol{\Sigma}$.

As described in [34], in the Guidard and Fischer cost function, there are two background terms; the standard background term

$$J_b(\boldsymbol{\delta x}) = \frac{1}{2}\boldsymbol{\delta x}^T \boldsymbol{B^{-1}}\boldsymbol{\delta x},$$

where $\boldsymbol{\delta x} = \boldsymbol{x} - \boldsymbol{x^b}$, and an extra term J_k. This extra term J_k measures the discrepancies from the large scales of the global analysis. In order to introduce a constraint with respect to the corresponding global analysis $\boldsymbol{x^{AA}}$, they add a new source of information $\mathcal{H}_1(\boldsymbol{x^{AA}})$, a subset of the global analysis. $\mathcal{H}_1$ is an operator bringing a vector from the global model space to a low-resolution LAM space, in order to constrain only the large scales in the LAM. Thus, $\mathcal{H}_1(\boldsymbol{x^{AA}})$ represents some large scales of the global analysis over the LAM domain.

The J_k term is given by

$$J_k(\boldsymbol{\delta x}) = \frac{1}{2}\left(\boldsymbol{d^k} - \boldsymbol{H_2}\boldsymbol{\delta x}\right)^T \boldsymbol{V^{-1}}\left(\boldsymbol{d^k} - \boldsymbol{H_2}\boldsymbol{\delta x}\right),$$

where $\boldsymbol{d^k} = \mathcal{H}_1(\boldsymbol{x^{AA}}) - \mathcal{H}_2(\boldsymbol{x^b})$ are the innovations with respect to the global large scales, $\boldsymbol{V}$ is the matrix of global-large-scale-error covariances and $\boldsymbol{H_2}$ is the linearisation of $\mathcal{H}_2$, where $\mathcal{H}_2$ is the an operator bringing a vector from the nominal-resolution LAM space to the same low-resolution LAM space as $\mathcal{H}_1$.

There appear to be similarities between the methods of both Guidard and Fischer, Ballard et al and ourselves. However, there are also clear differences. In the Ballard et al method, the large scales are constrained by the large scales in the parent analysis and the small scales are constrained by the small scales in the LAM background. In the Guidard and Fischer method, the large scales are constrained by both the large scales in the parent analysis and the large scales in the LAM background. The small scales are constrained by the small scales in the LAM background. In our method, the large scales are constrained by the large scales in the parent analysis and the small scales are (extremely weakly) constrained by the small scales in the parent analysis. However, the small scales are zero in the parent analysis.

Another difference between both these methods and the method using $\mathbf{\Sigma}$ developed in this thesis is that both the Guidard and Fischer and the Ballard et al methods have thus far only been used with 3D-Var, where as our method has been tested using 4D-Var. However, there is nothing in our method that is dependant on it being used with 4D-Var and it could easily be run for a 3D-Var scheme. Doing this would enable a direct comparison of the methods, which would be useful future work.

9.5 Summary

We have illustrated how choices of the matrix $\mathbf{\Sigma}$ relate to the corresponding matrix $\boldsymbol{B}$ in physical space. By placing a small variance on the low wavenumbers and a large variance on the high wavenumbers we generate a matrix $\boldsymbol{B}$ which oscillates and spreads information on the small scales. In contrast, if we place a large variance on the low wavenumbers and a small variance on the high wavenumbers we get the more traditional matrix $\boldsymbol{B}$ which has a smoother, wider form and spreads information on the large scales.

The correspondence between $\mathbf{\Sigma}$ and $\boldsymbol{B}$ was promising so we tested some simple choices of $\mathbf{\Sigma}$ in our model. Initial tests confimed that by putting a different variance on the large and small scales, $\mathbf{\Sigma}$ could split the scales in the analysis.

Further tests were done in a more realistic setting where the observations had added random noise and a background was taken to be the same as the reference trajectory but without the small scale information. This was done both with and without adding random noise to the background. Comparisons between analyses generated with different choices of $\mathbf{\Sigma}$ confirmed that varying the variances on the different wavenumbers allows the scales to be affected seperately in the data assimilation.

From the results of Chapter 8 we know that the LAM over estimates the low wavenumbers due to the LBCs and that the long-wave information is also aliased to the low wavenumbers. It was also shown that the parent analysis can be more accurate at the larger scales. Here we showed that by placing a smaller variance on the lower wavenumbers we can not only constrain the large scales so as to not degrade them in the LAM analysis, we can actually improve them compared to the background. At the same time we can accurately represent the small scales by balancing the variance on the higher wavenumbers in the background with that on the observations.

We also compared our methods to the work done by Guidard and Fischer [34] and Ballard et al [2]. It would appear, at least on initial inspection, that all three methods are using the same large scale information from the background to benefit the LAM analysis. It is just the method with which this information is absorbed that differs. This comparison of the methods warrants further study.

Chapter 10

Conclusions and future work

Hazardous or extreme weather is often caused by convective scale features. The need to predict these events accurately has created a growing need in NWP to improve our ability to forecast on the convective scale. To accurately predict the convective scale we need high resolution forecast models, in order to resolve the dominant motions correctly. However, this high resolution means the model can only have a limited area domain size. This causes problems for both the model and the data assimilation.

In Chapter 4 we introduced problems caused by the limited area nature of storm-scale forecast models: the need for lateral boundary conditions and the problem of capturing atmospheric phenomena with length scales longer than the domain of the limited area model, refered to in this thesis as long-waves, as well as small scale behaviour. Investigating the problems caused by the limited area domain has been the focus of this thesis.

To generate an accurate forecast on the convective scale it is important that a LAM analysis be able to capture all the different scales, both small and large, including those longer than the domain of the model. We investigated which scales can be captured by a LAM analysis and where information of these scales is projected. We also demonstrated a possible method to use this knowledge to improve the representation of the low wavenumbers in a LAM 4D-Var analysis whilst still maintaining the accurate representation of smaller scales. We now summarise the results and discuss ideas for future work.

10.1 Conclusions

We began in Chapter 2 by introducing data assimilation. We discussed variational techniques, in particular 4D-Var. In Chapter 4 we introduced the limited area models used operationally and considered some of the problems caused for both the model and the data assimilation by only having a limited domain size.

Chapters 2 and 4 provided the information necessary to set up a 4D-Var scheme on both a parent and a nested LAM domain. This was utilised in Chapter 5 where we outlined a general LAM domain and discussed the modifications necessary for performing 4D-Var on a LAM domain. The data assimilation was discussed for both a physical space (gridpoint) and spectral space scheme.

We introduced the DFT and the power spectrum in Chapter 3 as a method of understanding the analysis generated by the 4D-Var. The effect of the LAM domain on the DFT was examined in Chapter 6 and the sine transform was introduced. We demonstrated that long-waves are aliased onto other wavenumbers contained by the spectrum of the LAM. With the DFT the majority of the long-wave information is aliased onto wavenumber $k = 0$ and with the sine transform onto $k = 1$.

In Chapter 7 we introduced the 1D-advection-diffusion model as the model to be used in this thesis. Using information from Chapter 5 we set up both a gridpoint and spectral 4D-Var scheme on our LAM domain.

In Chapters 8 and 9 we used the model and 4D-Var schemes outlined in Chapter 7 to investigate how different scales are treated by the data assimilation. Using this knowledge we have developed a new method to improve the different scales in a LAM 4D-Var analysis using the background error covariance matrix $\mathbf{\Sigma}$. By controlling the variances in matrix $\mathbf{\Sigma}$ we can improve the representation of the larger scales whilst also maintaining the increased accuracy in the small scales achieved by the higher resolution of the LAM.

The main conclusions of Chapters 8 and 9 are summarised below;

- The higher resolution of the LAM data assimilation allows it to capture small scale detail that cannot be resolved by the parent model. However, to achieve this improvement in the small scales there must be high resolution observations available to the LAM data assimialtion.

- Even with perfect observations and no background term in the cost function, the LAM analysis cannot correctly capture the truth due to inaccuracies introduced through the boundary conditions.

- Errors introduced through the lateral boundary conditions, as well as limitations in the resolution of the LAM, cause the LAM data assimilation to have errors at low wavenumbers.

- In the LAM data assimilation, observational information of waves with lengthscales longer than the domain of the LAM is aliased onto other wavenumbers which are contained in the spectrum of the LAM, with the majority of the information being aliased onto the low wavenumbers in the LAM spectrum.

- Comparing the gridpoint and spectral schemes, it was shown that the truncation of high order wavenumbers in the finite spectral transform does not cause the spectral assimilation scheme to produce different results to the gridpoint scheme.

- It is important to generate the background trajectory $\boldsymbol{x_i^b}$ at the LAM resolution, and not simply interpolate a low resolution trajectory.

- Length scales in the 4D-Var analysis can be affected separately by the choice of variances in the background error covariance matrix $\mathbf{\Sigma}$.

- By over-constraining the low wavenumbers, we can not only stop them being degraded in the LAM analysis, we can actually improve them compared with the background. At the same time we can accurately represent the small scales by balancing the variances on the higher wavenumbers in the background with those on the observations.

The new method introduced in this thesis, which uses the matrix $\mathbf{\Sigma}$ to control the different scales, constraining the low wavenumbers to match the background while allowing the high resolution observations to benefit the small scales, could be applied to current operational methods which already implement a spectral control variable transform.

The matrix $\mathbf{\Sigma}$ method has shown promise here and warrants further study. However, the results generated also have some limitations.

By using a linear 1D-advection-diffusion model we were able to consider the different aspects of our problem in isolation. We investigated the effects of resolution and lateral boundary conditions, as well as the different length scales in separate experiments, making it much easier to identify which aspect was causing a particular outcome. This was possible because of the simple nature of our model. We also had an analytic solution with which to compare our model outputs.

As an initial test model the advection-diffusion model worked well. However, the discretisation scheme used suffered from numerical diffusion [62], being more diffusive at lower resolutions. This caused the LAM analysis to over estimate amplitudes at the inital time. Observing the effect of this error was interesting as all NWP models suffer from model error. However, to apply the method developed in this thesis operationally these results would need repeating for a different scheme, to quantify the errors.

Before the results of this thesis could be considered for operational applications they also need to be tested on a model with more of the properties of a full atmospheric model. Nonlinearity in particular, is an important property of atmospheric models, especially at the convective scale [19]. All of our results have been generated with a linear model. It is less clear whether the splitting of the scales using matrix $\mathbf{\Sigma}$ could be done so easily in a nonlinear model, as the wavenumbers could have more dependancy on each other. This requires further study before the method could be applied operationally.

Another limitation is that the assumption that the parent analysis accurately captures the larger scales may not always hold. However, this assumption is already made operationally when zero boundary conditions are imposed on the LAM [2]. Even if the large scales in the parent analysis are wrong for some reason, it may still prove beneficial to constrain the large scales in the LAM analysis to match those in the parent analysis. Given that the boundary conditions for the LAM are forced to remain equal to those provided by the parent, by constraining the large scales in the LAM analysis to match those in the parent analysis, the LAM is more consistent with the boundary conditions. This creates a smoother transition from the boundaries to the LAM interior and could reduce errors at the boundaries, such as ‘cut-off’ (as described in Section 4.2.1).

10.2 Further work

As discussed in Section 10.1, before our results using matrix $\mathbf{\Sigma}$ could be applied operationally they need to be tested in a nonlinear case. A useful next step would therefore be to incorporate some nonlinearity into our model. One possibility would be to develop our advection-diffusion model into a model of the 1D-Kuramoto-Sivashinsky (K-S) equation [69]. This equation contains the advection and the diffusion term we already have, but also contains a higher order spatial derivative and a nonlinear term. It exhibits self-sustained chaotic behaviour of a multiscale nature and has previously been used as a test model for 4D-Var on multiscale systems by [69]. The K-S model could be used to investigate whether nonlinearity affects the aliasing of the long-wave information, as well as whether the scales can still be split using matrix $\mathbf{\Sigma}$ when there is a nonlinear dependence between the wavenumbers.

A second extension to this work would be a more direct comparison between the matrix $\mathbf{\Sigma}$ method developed in this thesis, the J_k term proposed by Guidard and Fischer [34] and the scale-splitting proposed by Ballard et al [2]. As discussed in Chapter 9, all three methods are using large scale information from a parent analysis to benefit the LAM analysis. It is the method with which this information is absorbed that differs. Another difference is the small scale detail. Unlike both the other methods, our matrix $\mathbf{\Sigma}$ method does not at any point use a background provided by a LAM forecast. This means there is no small scale detail coming from the background. All the small scale detail is provided by the observations. The impact of this on the accuracy of the LAM analysis compared with those generated using the other methods would be useful to consider in future work.

If it did prove that the lack of high resolution detail in the background was detrimental to the accuracy of the LAM analysis, it would also be useful future work to consider a possible amendment to the background used in our method. A possible improvement could be to combine the large scales from a parent analysis with the small scales from a previous LAM forecast, and then use this combined state as the background for the LAM data assimilation. By splitting the scales in the background we would provide both large and small scale background information to the data assimilation. However, by splitting the scales in the background, and not in the analysis itself (as done in the

Ballard et al method [2]), we avoid the risk of an analysis that no longer satisfies the model equations.

Another useful next step would be to perform a direct comparison of the methods. A detailed mathematical analysis of the methods could provide an insight into their apparent similarities and provide a more formal comparison between them. Currently, both the Guidard and Fischer [34] and Ballard et al [2] methods have only been tested with 3D-Var. The mathematical formulation of the J_k term in a 4D-Var scheme has not been considered. However, the matrix $\mathbf{\Sigma}$ method developed in this thesis with 4D-Var, is equally applicable to 3D-Var, so 3D-Var could be used for initial comparison tests. If the three methods do produce similar results then the matrix $\mathbf{\Sigma}$ method provides an advantage as it does not need to be reformulated for use with 4D-Var.

Bibliography

[1] Arfken G. Mathematical Methods for Physicists. 3rd Edition. Academic Press Inc., 1985.

[2] Ballard S., Z. Li, M. Dixon, S. Swarbrick, O. Stiller and H. Lean. Development of 1-4km resolution data assimilation for nowcasting at the Met Office. World Weather Research Program Symposium on Nowcasting and Very Short Range Forecasting (WSN05) 2005; paper 3.02.

[3] Ballard S.P. Personal Communication.

[4] Bailey D. The Heat Equation, Simple Models, available at http://darc.nerc.ac.uk. Last accessed 27.01.2009.

[5] Bannister R. DARC INTERNAL REPORT NO. 5: On control variable transforms in the Met Office 3d and 4d Var., and a description of the proposed waveband summation transformation. Data Assimilation Research Centre, Department of Meteorology, University of Reading, UK. Document started Feb 2003.

[6] Bannister R. A review of forecast error covariance statistics in atmospheric variational data assimilation. I: Characteristics and measurements of forecast error covariances. *Quarterly Journal of the Royal Meteorological Society* 2008; **134**:1951–1970.

[7] Bannister R. A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics. *Quarterly Journal of the Royal Meteorological Society* 2008; **134**:1971–1996.

[8] Barker D.M., W. Huang, Y.R. Guo, A.J. Bourgeois and Q.N. Xiao. A Three-Dimensional (3DVAR) Data Assimilation System For Use With MM5: Implementation and Initial Results. *Monthly Weather Review* 2004; **132**:897–914.

[9] Bennett L.J., K.A. Browning, A.M. Blyth, D.J. Parker and P.A. Clark. A review of the initiation of precipitating convection in the United Kindom. *Quarterly Journal of the Royal Meteorological Society* 2006; **132**:1001–1020.

[10] Bryan G.H., J.C. Wyngaard and J.M. Fritsch. Resolution requirements for the simulation of deep moist convection. *Monthly Weather Review* 2003; **131**:2394–2416.

[11] Bjerknes V. Dynamic meteorology and hydrography. Part II. Kinematics. New York: Carnegie Institute, Gibson Bros, 1911.

[12] Bouttier F. and P. Courtier. Data assimilation concepts and methods. ECMWF Meteorological Training Course Lecture Series. March 1999;

[13] Cizek V. Discrete Fourier Transforms and their Applications. Adam Hilger, 1986.

[14] Clark T.L. Lateral and upper boundary conditions. Numerical Methods in Atmospheric Models, Seminar Procedings, ECMWF. 1991; Volume II:43–71.

[15] Cőté J., S. Gravel, A. Méthot, A. Patoine, M. Roch and A. Staniforth. The operational CMC-MRB global environmental multiscale (GEM) model. Part I: Design considerations and formulation. *Monthly Weather Review* 1998; **126**:1373–1395.

[16] Courtier P., J.-N. Thépaut and A. Hollingsworth. A stategy for operational implementation of 4D-Var, using an incremental approach. *Quarterly Journal of the Royal Meteorological Society* 1994; **120**:1367–1387.

[17] Courtier P. Dual formation of four-dimensional variational assimilation. *Quarterly Journal of the Royal Meteorological Society* 1997; **123**:2449–2461.

[18] Daley R. Atmospheric Data Analysis. Cambridge University Press, 1991.

[19] Dance S.L. Issues in high resolution limited area data assimilation for quantitive precipitation forecasting. *Physica D* 2004; **196**:1–27.

[20] Davies H.C. A lateral boundary formulation for multi-level prediction models. *Quarterly Journal of the Royal Meteorological Society* 1976; **102**:405–418.

[21] Davies H.C. Limitations of Some Common Lateral Boundary Schemes used in Regional NWP Models. *Monthly Weather Review* 1983; **111**:1002–1012.

[22] Dixon M., Z. Li, H. Lean, N. Roberts and S. Ballard. Impact of Data assimilation on forecasting convection over the United Kingdom using a high-resolution version of the Met Office Unified Model. *Monthly Weather Review* 2009; **137**:1562–1584.

[23] Durran D.R. Numerical Methods for Wave Equations in Geophysical Fluid Dynamics. Springer, 1999.

[24] Ehrendorfer M. Four-dimensional data assimilation: comparison of variational and sequential algorithms. *Quarterly Journal of the Royal Meteorological Society* 1992; **118**:637–713.

[25] Elliot D.F. and K.R. Rao. Fast Transforms. Algorithms, Analyses, Applications. Academic Press Inc., 1982.

[26] Fischer C., T. Montmerle, L. Berre, L. Auger and S.E. Ştefănescu. An overview of the variational assimilation in the ALADIN/France numerical weather-prediction system. *Quarterly Journal of the Royal Meteorological Society* 2005; **131**:3477–3492.

[27] Fox-Rabinovitz M., J. Ct, B. Dugas, M. Dqu and J.L. McGregor. Variable resolution general circulation models: Stretched-grid model intercomparison project (SGMIP). *Journal of Geophysical Research* 2006; **111**: D16104.

[28] Gauthier P. and J.-N. Thépaut. Impact of the digital filter as a weak constraint in the preoperational 4DVAR assimilation system of Météo-France. *Monthly Weather Review* 2001; **129**:2089–2102.

[29] Giering R. and T. Kaminski. Recipes for adjoint code construction. *Max-Planck-Institute for Meteorology* 1996; Technical Report 212.

[30] Glossary of Meteorology. AMS. 2nd Edition. Online version. http://amsglossary.allenpress.com/glossary. Last accessed 18.03.2009.

[31] Goldfarb D. A family of variable-metric methods derived by variational means. *Mathematics of Computation* 1970; **24**:23–26.

[32] Golding B., P. Clark and B. May. The Boscastle Flood: Meteorological analysis of the conditions leading to flooding on 16 August 2004. *Weather* 2005; **60(8)**:230–235.

[33] Golub G.H. and C.F. Van Loan. Matrix Computations. 3rd Edition. The John Hopkins University Press, 1996.

[34] Guidard G. and C. Fischer. Introducing the coupling information in a limited area variational assimilation. *Quarterly Journal of the Royal Meteorological Society* 2008; **134**:723–735.

[35] Gustafsson N., L. Berre, S. Hörnquist, X.-Y. Huang, M. Lindskog, B. Navascués, K.S. Mogensen and S. Thorsteinsson. Three-dimensional variational data assimilation for a limited area model. Part I: General formulation and the background error constraint. *Tellus* 2001; **53A**:425–446.

[36] Hand W.H., N.I. Fox and C.G. Collier. A study of twentieth-century extreme rainfall events in the United Kingdom with implications for forecasting. *Journal of Meteorological Applications* 2004; **11**:15–31.

[37] Hesthaven J.S., S. Gottlieb and D. Gottlieb. Spectral methods for time-dependent problems. Cambridge University Press, 2007.

[38] HIRLAM-5 scientific documentation. Dec 2002. Available at http://hirlam.org/. Last accessed 14.05.2009.

[39] General description of the HIRLAM model. Available at http://hirlam.org/ Last accessed 14.05.09.

[40] General description of the HARMONIE model. Available at http://hirlam.org/ Last accessed 14.05.09.

[41] Honda Y., M. Nishijima, K. Koizumi, Y. Ohta, K. Tamiya, T. Kawabata and T. Tsuyuki. A pre-operational variational data assimilation system for a non-hydrostatic model at the Japan Meteorological Agency:Formulation and preliminary results. *Quarterly Journal of the Royal Meteorological Society* 2005; **131**:3465–3475.

[42] Houtekamer P.L. and H.L. Mitchell. A sequential ensemble Kalman filter for atmosphereic data assimilation. *Monthly Weather Review* 2001; **129**:123–137.

[43] Huang X.-Y., X. Yang, N. Gustafsson, K.S. Mogensen and M. Lindskog. Four-dimensional variational data assimilation for a limited area model. Part I: General

formulation and the background error constraint. HIRLAM Technical Report **57**. Dec 2002; Available at http://hirlam.org/. Last accessed 14.05.2009.

[44] Ide K., P. Courtier, M. Ghil and A.C. Lorenc. Unified notation for data assimilation: Operational, sequential and variational. *J. Met. Soc. Japan* 1997; **75**:181–189.

[45] Isaacson E. and H.B. Keller. Analysis of numerical methods. John Wiley & Sons Inc., 1966.

[46] JMA. Outline of the operational numerical weather prediction at the Japan Meteorological Agency. 2007. Available from http://www.jma.go.jp/jma/jma-eng/jma-center/nwp/outline-nwp/index.htm. Last accessed 16.03.2009.

[47] Johnson C. Information Content of Observations in Variational Data Assimilation. PhD thesis, Department of Meteorology, University of Reading, 2003.

[48] Johnson L.W. and R.D. Riess. Numerical analysis. Addison-Wesley Publishing Company Inc., 1977.

[49] Kalman R.E.. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of basic engineering (Series D)* 1960; **82**:35–45.

[50] Kalnay E. Atmospheric Modeling, Data Assimilation and Predictability. Cambridge University Press, 2003.

[51] Lean H.W., P.A. Clark, M. Dixon, N.M. Roberts, A. Fitch, R. Forbes and C. Halliwell. Characteristics of high-resolution versions of the Met Office Unified Model for forecasting convection over the United Kingdom. *Monthly Weather Review* 2008; **136**:3408–3424.

[52] Leith C.E. Numerical Models of Weather and Climate. *Plasma Phys. Control. Fusion* 1993; **35**:919–927.

[53] Li Y, I.M. Navon, W. Yang, X. Zou, J.R. Bates, S. Moorthi and R.W. Higgins. Four-dimensional variational data assimilation with a multilevel semi-Lagrangian semi-implicit general circulation model. *Monthly Weather Review* 1992; **120**:1433–1446.

[54] Lorenc A.C. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society* 1986; **112**:1177–1194.

[55] Lorenc A.C. Development of an operational variational assimilation scheme. *Journal of the Meteorological Society of Japan* 1997; **75**:339–346.

[56] Lorenc A.C., 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* 2000; **126**:2991–3012.

[57] Lorenc A.C. and F. Rawlins. Why does 4D-Var beat 3D-Var? *Quarterly Journal of the Royal Meteorological Society* 2005; **131**:3247–3257.

[58] Lorenc A.C. and T. Payne. 4D-Var and the butterfly effect: Statistical four-dimensional data assimilation for a wide range of scales. *Quarterly Journal of the Royal Meteorological Society* 2007; **133**:607–614.

[59] Lorenz E.N. The Essence of Chaos. UCL Press, 1993.

[60] Marsham J. Meeting report: Mid-latitude convective storms and their initiation. *Weather* 2006; **61(6)**:175–177.

[61] Met Office. Atmospheric numerical model configurations. Available at http://www.metoffice.gov.uk/science/creating/daysahead/nwp/um_config.html. Last accessed 18.03.2009.

[62] Morton K.W. Numerical solution of convection-diffusion problems. Chapman & Hall, 1996.

[63] Navon I.M., X. Zou, J. Derber and J. Sela. Variational data assimilation with an adiabatic version of the NMC spectral model. *Monthly Weather Review* 1994; **122**:966–983.

[64] Nichols N.K. Data assimilation: Aims and basic concepts. In *Data Assimilation for the Earth System,* Swinbank R, Shutyaev V, Lahoz WA (eds). Kluwer Academic: 2003; 9–20.

[65] Park S.K. and D. Zupanski. Four-dimensional variational data assimilation for mesoscale and storm-scale applications. *Meteorology and Atmospheric Physics* 2003; **82**:173–208.

[66] Pielke R.A. Mesoscale Meteorological Modeling. Academic Press, 2002.

[67] Pires C., R. Vautard and O. Talagrand. On extending the limits of variational assimilation in nonlinear chaotic systems. *Tellus* 1996; **48A**:96–121.

[68] Press W.H., S.A. Teukolsky, W.T. Vetterling and B.P. Flannery. Numerical Recipes in FORTRAN. The art of scientific computing. 2nd Edition. Cambridge University Press, 1992.

[69] Protas B., T.R. Bewley and G. Hagen. A computational framework for the regularization of adjoint analysis in multiscale PDE systems. *Journal of Computational Physics* 2004; **195**:49–89.

[70] Rabier F., P. Courtier, J. Pailleux, O. Talagrand and D. Vasiljevic. A comparision between four-dimensional variational assimilation and simplified sequential assimilation relying on three-dimensional variational analysis. *Quarterly Journal of the Royal Meteorological Society* 1993; **119**:845–880.

[71] Rabier F., H. Järvinen, E. Klinker, J.-F. Mahfouf and A. Simmons. The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. *Quarterly Journal of the Royal Meteorological Society* 2000; **126**:1143–1170.

[72] Ramirez R.W. The FFT Fundamentals and Concepts. Prentice-Hall, 1985.

[73] Reid J.K. On the method of conjugate gradients for the solution of large systems of linear equations. Large sparse sets of linear equations: Proceedings of the Oxford conference of the Institute of Mathematics and its Aplications held in April, 1970. Academic Press, 1971.

[74] Roberts N. Meteorological components in forecasts of extreme convective rainfall using 12-km and 1-km NWP models: A tale of two storms. Met Office Forecasting Research Technical Report **500**, Joint Centre for Mesoscale Meteorology Report **157**, April 2007; Available from the Met Office website http//www.metoffice.gov. Last accessed 14.05.2009.

[75] Roberts N.M. and H.W. Lean. Scale-selective verification of rainfall accummulation from high-resolution forecasts of convective events. *Monthly Weather Review* 2007; **136**:78–97.

[76] Shanno D.F. and K.H. Phau. The CONMIN code. Minimisation of unconstrained multivariate functions. *ACM Transactions on mathematical software 6* 1980; 618–622. Available at http://gams.nist.gov/serve.cgi/Module/TOMS/500/8535. Last accessed 30.01.09

[77] Shanno D.F. Conditioning of quasi-Newton methods for function minimisation. *Mathematics of Computation* 1970; **24**:647–656.

[78] Skamarock W.C., J.B. Klemp, J. Dudia, G.O. Gill, D.M. Barker, M.G. Duda, X-Y Huang, W. Wang and J.G. Powers. A description of the Advenced Reasearch WRF Version 3. NCAR Technical Note **475** 2008; Available from http://www.wrf-model.org. Last accessed 14.05.2009.

[79] Staniforth A., A. White, N. Wood, J. Thuburn, M. Zerroukat, E. Cordero, T. Davies et al. Joy of U.M. 6.1 - Model Formulation. Unified Model Documentation Paper No 15. February 2005; These are unpublished papers. Available from the Met Office website http//www.metoffice.gov.uk

[80] Strang G. Wavelet transforms versus Fourier transforms. *Bulletin of the American Mathematical Society* 1993; **28**:283–305.

[81] Sun J. Convective-scale assimilation of radar data: Progress and challenges. *Quarterly Journal of the Royal Meteorological Society* 2005; **131**:3439–3463.

[82] Trefethen L.N. and D. Bau. Numerical linear algebra. SIAM, 2007.

[83] Wan F.Y.M. Introduction to the Calculus of Variations and its Applications. Chapman & Hall Mathematics, 1993.

[84] Warner T.T., R.A. Peterson and R.E. Treadon. A tutorial on lateral boundary conditions as a basic and potentially serious limitation to regional numerical weather prediction. *Bulletin of the American Meteorological Society* 1997; **78**(11):2599–2617.

[85] Watkinson L. Four Dimensional Variational Data Assimilation for Hamiltonian Problems. PhD thesis, Department of Mathematics, University of Reading, 2006.

[86] Xue Ming, D. Wang, J. Gao, K. Brewster and K.K. Droegemeier. The Advanced Regional Prediction System (ARPS), storm-scale numerical weather prediction and data assimilation. *Meteorology and Atmospheric Physics* 2003; **82**:139–170.

[87] Zou X. and Y.H. Kuo. Rainfall assimilation through an optimal control of initial and boundary conditions in a limited-area mesoscale model. *Monthly Weather Review* 1996; **124**:2859–2882.