THE UNIVERSITY OF READING
DEPARTMENTS OF
MATHEMATICS AND METEOROLOGY

Automatic generation of accurate advection schemes on unstructured grids and their application to meteorological problems.

by Raymond K.Lashley

Dissertation submitted for the degree of
Doctor of Philosophy

June 2002

Abstract

A class of two-dimensional, conservative, shape preserving advection schemes is developed for use in meteorological models on any grid. The advection scheme is formulated in finite volume form to ensure conservation. A local high order polynomial reconstruction of the advected field is integrated over the region swept across each face to generate the fluxes across cell edges. A flux limiter is applied to these fluxes to ensure shape preservation of the advected profile. This procedure is generalised and automated to enable it to be used on any grid on the plane and on the sphere. The method is also included in a shallow water model on the sphere.

The automatic generation of the schemes also enables the order of the polynomial used to be easily changed to vary both the accuracy and the computational cost of the scheme. The effect on the accuracy and computational cost of varying the order of the polynomial and the grid resolution has been investigated on square and triangular grids on the plane, and the icosahedral-hexagonal grid on the sphere. It is found that second to fourth order polynomial are the most efficient when comparing a schemes accuracy with its computational cost.

The icosahedral-hexagonal grid gives a near uniform coverage of the sphere and has no singularities, which makes it particularly suitable for use in meteorological models. The schemes developed in this work have been incorporated into a global shallow water model on an icosahedral-hexagonal grid. A standard set of test cases for shallow water models were used with the model to investigate the effect of using more accurate advection schemes in the model. The use of higher order polynomials in the advection scheme produces significant improvements in the accuracy of the model without the increase in computational cost associated with increasing the grid resolution.

To explain all nature is too difficult a task for any one man or even for any one age. 'Tis much better to do a little with certainty, and leave the rest for others that come after you, than to explain all things.

Sir Isaac Newton

Acknowledgments

There are so many people that deserve thanks for lè o me t co pl ttsl by nh faults.

To everyone else, Scout leaders, co-workers, RUSU colleagues, fellow X-ers and so many more, thank you. However brief our passings the result was to put me on or help me along this path. Special thanks and apology to John Henderson, who told me the truth I didn't want to hear. It caused us both difficulties but I appreciate it more than anyone will know.

Final thanks to an unknown author of a pre-school maths book, a green one if I remember correctly, for making me want to get a PhD in Mathematics. This was all your fault, but at least I've got it out of my system now.

Contents

Chapter 1

Introduction

tions.

These approximations can be made either to the equations themselves or to the way the fields are represented in a model. Approximations to the equations may involve linearising and removing 'small' terms or assuming balance conditions that are only approximate in the real problem. There are also many ways that the problem can be discretised in both space and time. These approximations can be combined in many different ways for many different purposes, including studying individual atmospheric processes, forecast models and climate simulations.

Richardson [36] proposed 'solving' the equations describing the atmosphere by discretising them in space and stepping forward in time by using numerical approximations to the derivatives in the equations. Richardson's first forecast was not a success by todays standards for a number of reasons. Both the model equations and the methods used to solve them were rather basic approximations and the initial data was poorly incorporated, as suggested by Lynch [26]. Despite these initial difficulties, the idea has developed and is now used for forecasting as well as running idealised models of individual atmospheric processes and climate simulations.

Since the advent of computers, numerical methods have been refined as more and more calculations can be performed in shorter and shorter times (see Wiin-Nielsen [57]). These refinements have covered all aspects of the numerical approximations of the governing equations, including advection. The type of schemes that have been used for advection vary from the simple first order Euler scheme to high order linear and nonlinear schemes. Cullen [5] gives an overview of finite difference methods in a meteorological context whilst Rood [38] gives a review of many finite difference schemes for advection. Many of these schemes are one dimensional, linear schemes which equation (1.1) can also be written in advective form,

$$\frac{\mathrm{D}\phi}{\mathrm{D}t} = 0 \ , \tag{1.3}$$

where $\frac{\mathrm{D}}{\mathrm{D}t}$ is the Lagrangian derivative;

$$\frac{\mathrm{D}}{\mathrm{D}t} \equiv \frac{\partial}{\partial t} + \mathbf{v} \cdot \nabla \ . \tag{1.4}$$

This shows that the mixing ratio of a fluid parcel does not change as it is being advected. There can be no increase in the global maximum mixing ratio, or decrease in the minimum, and there would be no generation of new local maxima or minima.

Either or both of these properties, known as 'conservation' and 'shape preservation' (or monotonicity), can be imposed on numerical solutions by using suitable numerical schemes. Ensuring conservation means that our advection scheme will not create or destroy amounts of whatever is being advected. This is particularly important when the advected fields are interacting with one another and the balance between them is important e.g. the transport of reactive chemicals. Ensuring shape preservation means that oscillations will not be generated near sharp gradients in the field and that no new unphysical maxima or minima will be introduced. This ensures that fields of advected quantities remain positive and do not display unphysical oscillations.

Conservation can be easily ensured by using a finite volume type scheme (e.g. the UTOPIA scheme [23]) or by a more careful formulation of other finite difference or finite element schemes. Enforcing the shape preservation property can lead to more complicated numerical schemes. Godunov's theorem (see LeVeque [25]) showed that any linear, shape preserving Lenuh nuM mrty ill r, Lre thge ldf r cEnacy arp crotof ha ith . not linear and there are a number of methods of generating such schemes. Van Leer incorporated these features into his search for the ultimate conservative difference scheme [17, 18, 19, 20, 21]. The Universal Limiter [22, 51] uses local information about the advected field to set bounds on fluxes that ensure shape preservation. Suresh and Huynh [47] present a fifth order scheme with a limiter that can be applied with Runge-Kutta time stepping in one dimension.

Flux Corrected Transport (FCT) schemes [2, 3, 4, 61] take a similar approach using a first order, shape preserving scheme, and applying carefully constrained 'antidiffusive' fluxes. Total Variation Diminishing (TVD) schemes [48] enforce a property similar to shape preservation which may allow extrema to grow if a nearby one diminishes. Essentially Non-Oscillatory (ENO) schemes [8, 40] do not enforce the shape preservation but do ensure that the magnitude of spurious oscillations decays with increasing resolution.

Making improvements to a numerical solution of the advection equation can be done in two ways. Increasing the computer power devoted to the problem will enable more complex and more accurate schemes or higher resolution to be used. Improving the numerical methods that are used can also improve the numerical solutions obtained. Improving the efficiency of a scheme will allow measure of the error in a numerical solution, a more accurate scheme is one that gives a better (usually lower) value of this error. Different error measures can behave differently for the same numerical schemes, a simple example of this can be seen in the L-norms of the Laplacian operator in Majewski et al. [28] (Fig. 8). The L_1 and L_2 errors display a second order accuracy as the grid is refined but the L_∞ error is only first order. In this and many other cases, the 'order of accuracy' is used to describe the rate at which the accuracy of the scheme improves as the grid is refined.

It is a well known but often overlooked fact that a 'high order' scheme may have a lower accuracy than a 'low order' scheme at some resolution. Accuracy can also be measured in a qualitative sense whereby a particular property of the solution may be important as well as or instead of an error measure. For example, it may be important that the numerical solution of the advection equation does not contain any spurious oscillations, even if this means that an error measure in increased. In this work there are occasions when we can relate the order of the polynomial used by a scheme to the formal order of accuracy of the scheme and times when this is not possible. For this reason, reference to the 'order' of a scheme refers to the order of the polynomial used, the formal order of accuracy will be referred to explicitly when required.

The use of high order accurate advection schemes can have a significant improvement in meteorological models. An example of this is given by Gregory and West [7] who investigated the use of different schemes to model the 'tape-recorder signal' of water vapour in the stratosphere. This work showed that the signal propagated vertically much too fast when a second order scheme was used. When higher order accurate schemes were used there were strong unphysical oscillations in solutions, limiters were required to remove these oscillations.

existing numerical schemes cannot be used. The icosahedral-hexagonal grid is described in detail in chapter 4.

The aim of this work is to develop an accurate, conservative and shape preserving advection scheme that is both fully multidimensional and can be used on unstructured grids. The properties of the scheme are desirable for some of the reasons described above and because we hope to improve the results of the Shallow Water model on the icosahedral-hexagonal grid developed by Thuburn [52, 53]. This model used a multidimensional advection scheme that was conservative and shape preserving but was based on a method for use on a regular hexagonal grid. This scheme caused some problems when used with the icosahedral-hexagonal grid that we might expect a tailor made scheme not to reproduce.

Attempting to develop an advection scheme that produces a numerical solution with all the desired properties (accuracy, conservation, shape preservation) will require some sacrifices to be made. The approach taken here is to try and include these sacrifices within the scheme by developing a method that will generate a family of schemes. Conservation can be assured by discretising the problem into finite volumes and approximating the fluxes between the control volumes (cells). The method is designed to allow greater accuracy to be gained at the expense of computational cost. In common with many advection schemes we shall find a scheme for which the tracer value at the next time step, ϕ^{n+1} can be written in terms of a set of coefficients, c, and the tracer values at the current time step, ϕ^n. This means we can write down an update equation in the form

$$\phi_i^{n+1} = \sum_j c_j \phi^n$$

coefficients, c_j, that we seek to make up the advection scheme. If the method is to be applied to unstructured grids then it will be necessary for the method of finding these c's to be automated. This is because the different sizes and shapes of the cells prevent us from finding a general set of c's for all cases. If the schemes are used to find the fluxes of the advected quantity between the cells then shape preservation can be guaranteed by use of the Universal Limiter as described in Thuburn [51].

The method for generating these schemes is based on a polynomial interpolation of the advected field which is used to approximate the fluxes. This is the same idea as used by Leonard in his ULTIMATE scheme [22] in one dimension and UTOPIA scheme [23] in two dimensions. Chapter 2 begins by showing how automating the basic method used by the ULTIMATE schemes enables the same idea to be applied to non-uniform grids. Extending the ULTIMATE methodology to two dimensions enabled the development of the UTOPIA scheme. Using the idea of automatic generation allows this to be extended to use any order polynomial when generating the advection scheme, not just the second order polynomial[1] used in the UTOPIA scheme. The details of this extension on rectangular grids and the results of idealised test cases are also included in Chapter 2.

The method described in Chapter 2 cannot be applied directly to non-rectangular grids since it relies on properties of the connectivity of the grid. This means that the method must be generalised further to enable its application to other grids. A new way of fitting the polynomial is needed in order to enable the method to be applied to non-rectangular

---

[1] The UTOPIA scheme was derived using a third order polynomial for interpolation to generate an update equation. It was suggested by Leonard et al. [23] that the scheme could be written in terms of fluxes across cell boundaries that were analogous to the one dimensional quadratic interpolation schemes with extra two dimensional terms. It will be shown later in this work that the UTOPIA scheme is very similar to a scheme generated directly using a two dimensional second order polynomial interpolation

grids. Chapter 3 covers some of the ways this may be done in one dimension and goes on to apply the most suitable way to triangular grids in two dimensions. The results from idealised test cases similar to those in Chapter 2 are also presented.

Chapter 4 applies the method developed in Chapter 3 to a hexagonal-icosahedral grid on the sphere. This does not require any further changes to be made to the method itself but some extra approximations are made because of the spherical geometry. A description of these approximations and their effects in one dimension are given in Chapter 4 along with results of test cases from the application of this method to the spherical icosahedral grid.

In order to test if these schemes have any effect in more complex systems of equations they are used in a shallow water model. The results of the test cases proposed by Williamson et. al. are presented in Chapter 5. Chapter 6 discusses the conclusions that can be drawn from the previous chapters and discusses some of the points raised. This chapter also contains suggestions for further work to be done with this method in order to improve it further.

Chapter 2

Rectangular Grids

2.1 Introduction

In the last chapter several properties that are desirable in a numerical scheme were described, these included, conservation, shape preservation, multidimensionality and good accuracy. Advection schemes with these properties already exist, the ULTIMATE scheme [22] in one dimension and the UTOPIA scheme and Universal Limiter [23], but we will need to extend these methods for our purposes. The ULTIMATE family of schemes uses polynomial interpolation in one dimension, allowing greater accuracy to be achieved by increasing the order of the polynomial at the expense of increasing the computational cost. This ability to vary the accuracy and cost of the scheme is a useful feature that we would like to retain but the current method can only be applied to regular grids. We shall begin in Section 2.2 by automating the process of generating the advection scheme so we can apply the same method of the ULTIMATE schemes to irregular grids in one dimension.

The UTOPIA scheme uses a second order multidimensional polynomial in the same way as the ULTIMATE scheme does in one dimension. We can expand on this idea to use higher order multidimensional polynomials to increase the accuracy and the computational cost of the schemes. The methodology for doing this is covered in Section 2.3.1 and the results of using the schemes are presented in Section 2.4. Once we have achieved an automation of the process of generating advection schemes and extended the schemes to two dimensions we can go on to look at extending the methods to irregular grids.

2.2 One-Dimensional Advection

The advection equation, in one dimension, of a tracer with mixing ratio, ϕ, by a non-uniform velocity $u(t,x)$ can be written,

$$\frac{\partial \phi}{\partial t} = \frac{\partial\left(-u\phi\right)}{\partial x} \,. \tag{2.1}$$

The fluid density, ρ, is taken to be constant in this case so ϕ can be thought of as representing the average tracer amount in the cell. By integrating this equation over a time Δt and a control volume i in space we get the exact conservative form

$$\phi_i^{n+1} = \phi_i^n + \left(\hat{\phi}_{\mathcal{L}} - \hat{\phi}_{\mathcal{R}}\right), \tag{2.2}$$

where ϕ_i^n is the average mixing ratio in cell i at time $n\Delta t$ and $\hat{\phi}_{\mathcal{L}/\mathcal{R}}$ is the flux across the left/right edge of cell i in time Δt. The flux may be thought of as the average mixing ratio ϕ at the edge over time Δt, multiplied by the Courant number $\frac{u\Delta t}{\Delta x}$, the ratio between the mass of fluid swept across the edge and the total mass in the cell. It can also be thought of as the amount of the tracer swept across the edge during the time step, shown by the bold regions in Figure 2.1. It is these fluxes that we wish to approximate in order to produce an update equation for our numerical method, based on equation (2.2) .

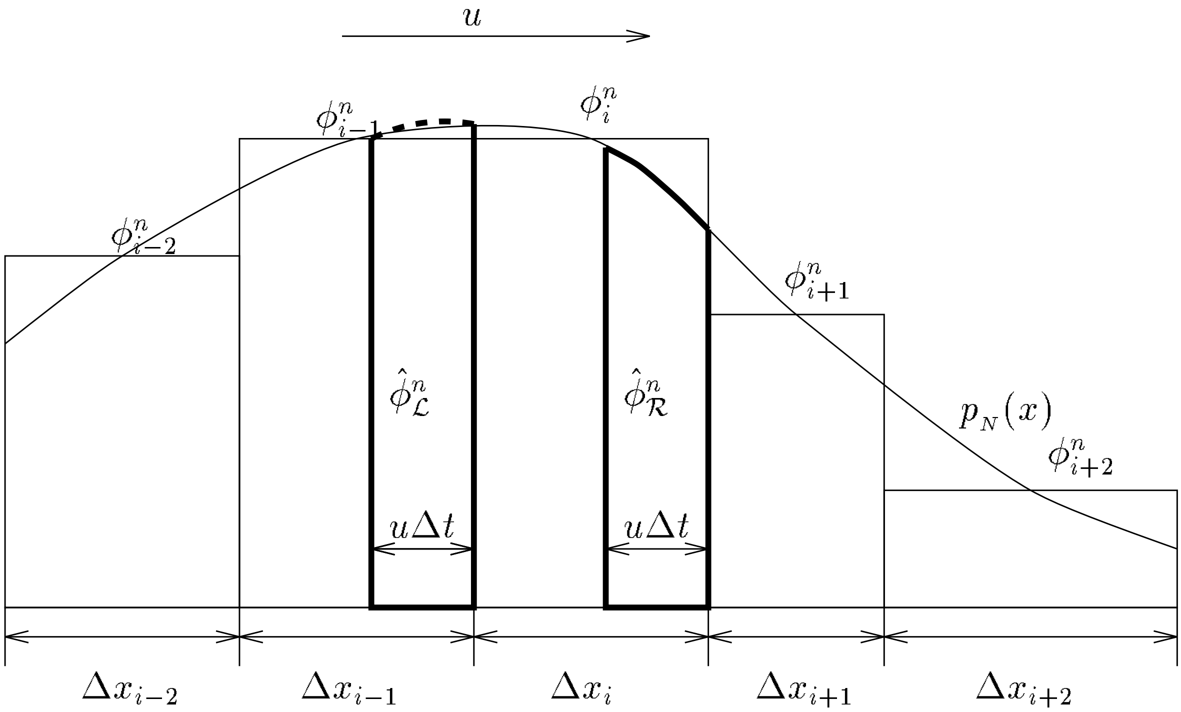


Figure 2.1: Diagram showing the regions used to calculate the fluxes, $\hat{\phi}^n_{\mathcal{L}/\mathcal{R}}$, using the polynomial, p_N. The polynomial P_N is interpolated over $N+1$ cells. The integral of this polynomial over the region swept across an edge in one time step is the approximation to the flux. A different polynomial will be found for each edge.

In his paper [22] Leonard derives update equations for advection of a tracer by a variable advecting velocity using polynomial approximations to the continuous tracer field. An N^{th} order polynomial is interpolated through $N+1$ cell centres spaced either side of the edge to approximate the tracer field. This polynomial can then be used instead of the unknown ϕ-distribution to approximate the fluxes in a similar way to that shown in Figure 2.1. This was done pointwise on a grid with a uniform spacing, Δx, but the method can be extended to a grid with unequal Δx's. For such a grid we would not wish to write down, and then code up, the update equation for each cell in turn. Instead we shall develop the method in such a way that it can easily be coded up to calculate the coefficients in the update equation automatically. This will give us an update equation

where N is the order of the polynomial being used. This polynomial has $N+1$ terms and so we insist that the above holds for $N+1$ distinct cells. This generates $N+1$ equations for the $N+1$ unknowns and thus uniquely defines a polynomial. The reason why this will not be a singular problem will be made clear when we investigate the error of the schemes. If $N+1$ is even then the nearest $\frac{N+1}{2}$ cells on each side of the edge are used for a centred scheme, otherwise the nearest $\frac{N}{2}$ cells downwind of the edge and $\frac{N}{2}+1$ cells upwind of the edge are used for an upwinded scheme.

If we define a local coordinate system with the origin at the edge for which we wish to approximate the flux, and write $p_{_N}(x)$ as $\sum_{j=0}^{N} a_j x^j$ then equation (2.4) can be written as

$$\Delta x_i \phi_i^n = \sum_{j=0}^{N} \int_{S_i}^{S_i+\Delta x_i} a_j x^j \mathrm{d}x \ , \tag{2.5}$$

where S_i is the distance of the left edge of cell i from the local origin. Integrating this gives

$$\phi_i^n = \frac{1}{\Delta x_i} \sum_{j=0}^{N} \frac{a_j}{j+1} \left\{ (S_i + \Delta x_i)^{j+1} - S_i^{j+1} \right\} \ . \tag{2.6}$$

Using the binomial expansion we can write this as

$$\phi_i^n = \sum_{j=0}^{N} \frac{a_j}{(j+1)\Delta x_i} \left(S_i^{j+1} + \sum_{k'=1}^{j+1} \left(\frac{(j+1)!\Delta x_i^{k'} S_i^{j+1-k'}}{(j+1-k')!k'!} \right) - S_i^{j+1} \right) \ . \tag{2.7}$$

Substituting $k' = k+1$ and simplifying gives

$$\phi_i^n = \sum_{j=0}^{N} a_j \sum_{k=0}^{j} \frac{j!\Delta x_i^k S_i^{j-k}}{(j-k)!(k+1)} \ . \tag{2.8}$$

By insisting that this holds for each cell in the stencil, we can write down the following $N+1$ equations for the unknown a_j's,

$$\phi_i^n = a_0 + \sum_{j=1}^{N} a_j \sum_{k=0}^{j} \frac{j!}{(j-k)!(k+1)!} \Delta x_i^k S_i^{j-k} \qquad , \ S_i \neq 0, \tag{2.9a}$$

$$\phi_i^n = a_0 + \sum_{j=1}^{N} \frac{a_j \Delta x_i^j}{j+1} \qquad , \ S_i = 0 \ . \tag{2.9b}$$

$S_i = 0$ for the cell to the right of the edge for which we are approximating the flux, i.e. cell i. The solution of these equations for the unknown a_j's (in vector $\mathbf{a}$) is a matrix inversion problem of the form

$$\begin{pmatrix} 1 & P_{1,i-\frac{N+1}{2}} & P_{2,i-\frac{N+1}{2}} & \cdots & \cdots & \cdots & P_{N,i-\frac{N+1}{2}} \\ \vdots & \vdots & \vdots & \ddots & & & \vdots \\ 1 & P_{1,i-1} & P_{2,i-1} & & & & P_{N,i-1} \\ 1 & \frac{\Delta x_i}{2} & \frac{\Delta x_i^2}{3} & & \ddots & & \frac{\Delta x_i^N}{N+1} \\ 1 & P_{1,i+1} & P_{2,i+1} & & & & P_{N,i+1} \\ \vdots & \vdots & \vdots & & & \ddots & \vdots \\ 1 & P_{1,i+\frac{N+1}{2}} & P_{2,i+\frac{N+1}{2}} & \cdots & \cdots & \cdots & P_{N,i+\frac{N+1}{2}} \end{pmatrix} \mathbf{a} = \boldsymbol{\phi} \ . \tag{2.10}$$

The first or last row, depending on the flow direction, will not be included if an even order polynomial is used (i.e. $N+1$ is odd). $P_{n,m}$ is a two dimensional polynomial in Δx_m and S_m consisting of all terms of order n, taken from equation (2.9a);

$$P_{n,m} = \sum_{k=0}^{n}$$ [illegible]

out a diagonal matrix of powers of a general Δx

c's from equation (2.3),

$$c_{i_{\mathcal{L}}} = \begin{pmatrix} \frac{-\mathsf{B}^{-1}_{i0}}{1} \\ \frac{-\mathsf{B}^{-1}_{i1}}{2} \\ \vdots \\ \frac{-\mathsf{B}^{-1}_{iN}}{N+1} \end{pmatrix} \cdot \begin{pmatrix} 1 \\ -\alpha \\ \vdots \\ (-\alpha)^N \end{pmatrix} \qquad , i = 0, 1, \ldots, N \; . \tag{2.17}$$

If the flow is constant in time then $\boldsymbol{c}$ can be calculated once for each edge, otherwise this matrix-vector product will need recomputing each time the flow changes at that edge.

Using this method, we can calculate the flux at each edge as the dot product of a vector of coefficients with a vector of tracer values in cells neighbouring the edge. The vector of coefficients is constant at an edge in equation (2.3) if the flow across that edge is constant, otherwise it can be formed from a matrix-vector product. This vector is dependent on the speed and direction of the flow and the matrix is dependent only on the direction of the flow and the grid. In order to compute this method efficiently, we first form the matrix B for each edge and calculate its inverse. If this is the first time step or if the flow has changed at an edge, then the product of the corresponding B^{-1} and $\boldsymbol{\alpha}$ is formed. The dot product of this vector, or one calculated at a previous time step, with the corresponding $\boldsymbol{\phi}$ gives the flux.

This enables coefficients of α and the ϕ_i's to be calculated for all cells. The coefficients of α depend on the grid size and so need only be calculated once for a regular grid. When this is the case the coefficients of ϕ depend only on α and need only be calculated once for uniform, constant flow. These coefficients are used to calculate the fluxes for the update equation which are in turn used to approximate the tracer distribution at the next time step. This last step is repeated for each time step.

step. As would be expected, the more accurate the interpolation, the smaller the error will be.

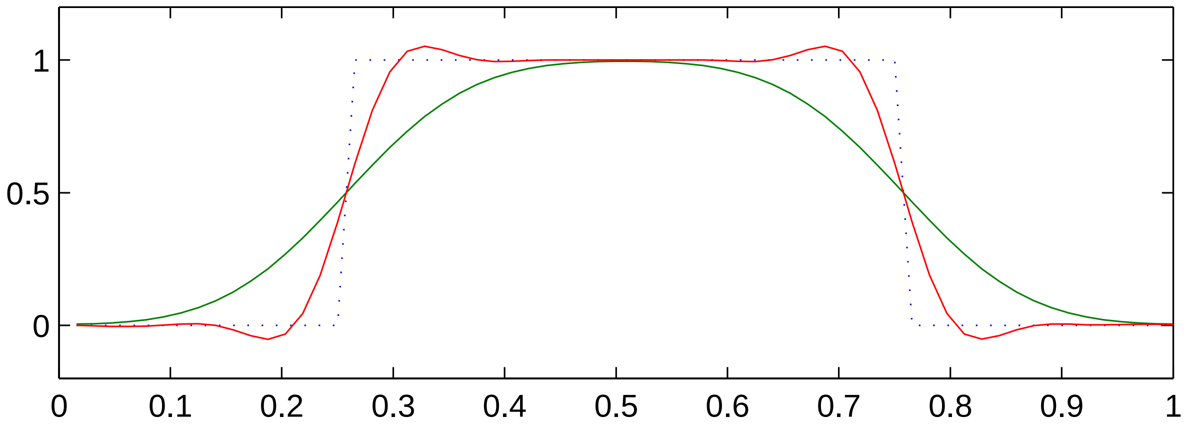
The polynomial is fitted by insisting that

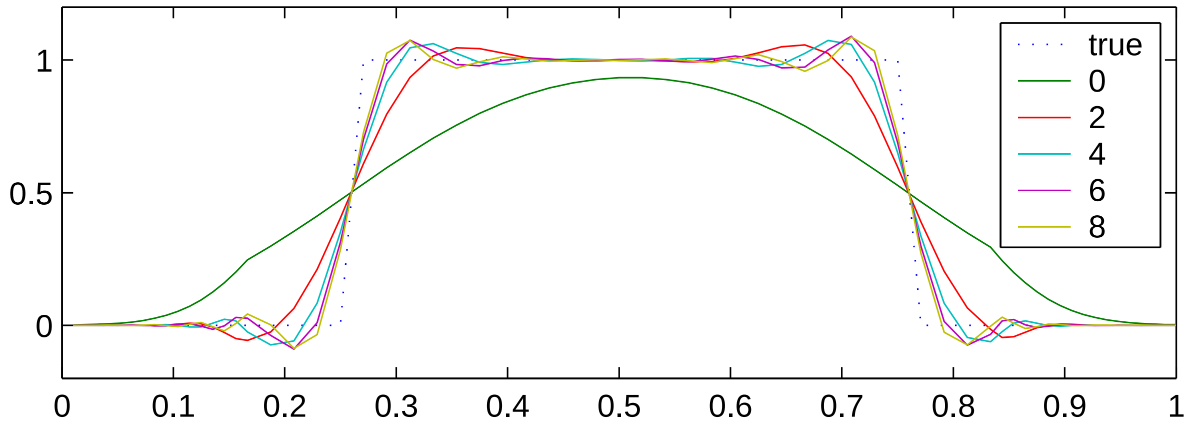
$$\Delta x_i \phi_i^n = \int_{S_i}^{S_{i+1}} p_{_N}(x) \mathrm{d}x \ , \tag{2.23}$$

over $N+1$ cells. The right hand side of this equation can be re-written, using the mean value theorem, as

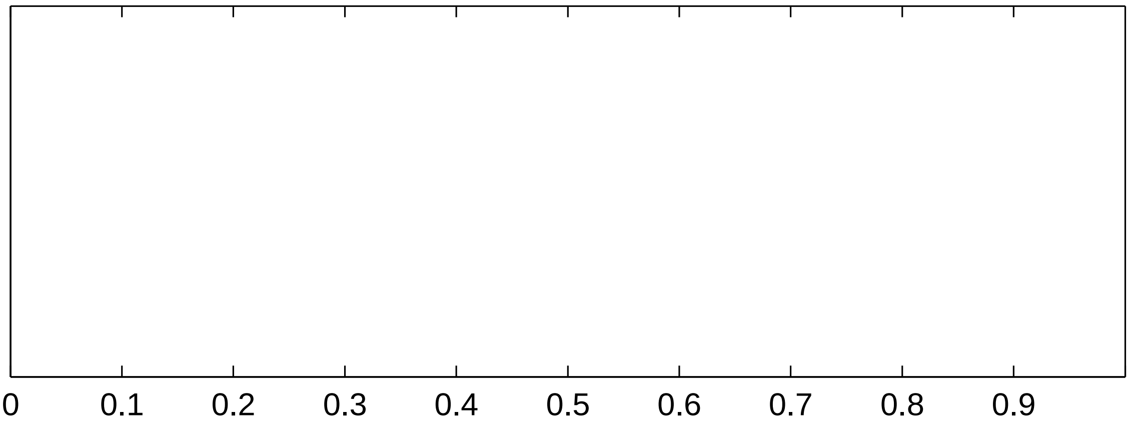
$$\Delta x_i \phi_i^n = \Delta x_i p_{_N}(x_\zeta) \ , \tag{2.24}$$

where $S_i < x_\zeta < S_{i+1}$. Written in this form we can clearly see that the interpolation of equation (


1
0.5
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1



1
0.5
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
true
0
2
4
6
8



0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

2.3 Two-Dimensional Advection

We can attempt to extend the same method into two dimensions on a rectangular grid simply by using two-dimensional polynomials and integration. Another way to apply the method in two dimensions would be to use dimensional splitting. This would involve applying the one-dimensional method separately in each coordinate direction. This has been shown by Leonard et al. [23] and Thuburn [51] to cause distortion performing this integration gives

$$\phi_{ij}^{n} = \sum_{k=0}^{N} \sum_{l=0}^{k} \frac{a_z}{\Delta x \Delta y} \left(\frac{(i\Delta x)^{k-l+1}}{k-l+1} - \frac{((i-1)\Delta x)^{k-l+1}}{k-l+1} \right) \left(\frac{(j\Delta y)^{l+1}}{l+1} - \frac{((j-1)\Delta y)^{l+1}}{l+1} \right), \tag{2.27}$$

which can be simplified using the binomial expansion to give

$$\phi_{ij}^{n} = \sum_{k=0}^{N} \sum_{l=0}^{k} a_z \Delta x^{k-l} \Delta y^{l} \left(\sum_{m=0}^{k-l} \frac{(-1)^m (k-l)! i^{k-l-m}}{(k-l-m)! m!} \right) \left(\sum_{p=0}^{l} \frac{(-1)^p (l)! j^{l-p}}{(l-p)! p!} \right) . \tag{2.28}$$

By insisting that equation (2.28) holds over K cells we can set up a matrix problem of the form, $\mathsf{B}\mathbf{a} = \boldsymbol{\phi}$. By carefully selecting the stencil over which we fit this polynomial, we can en [illegible]

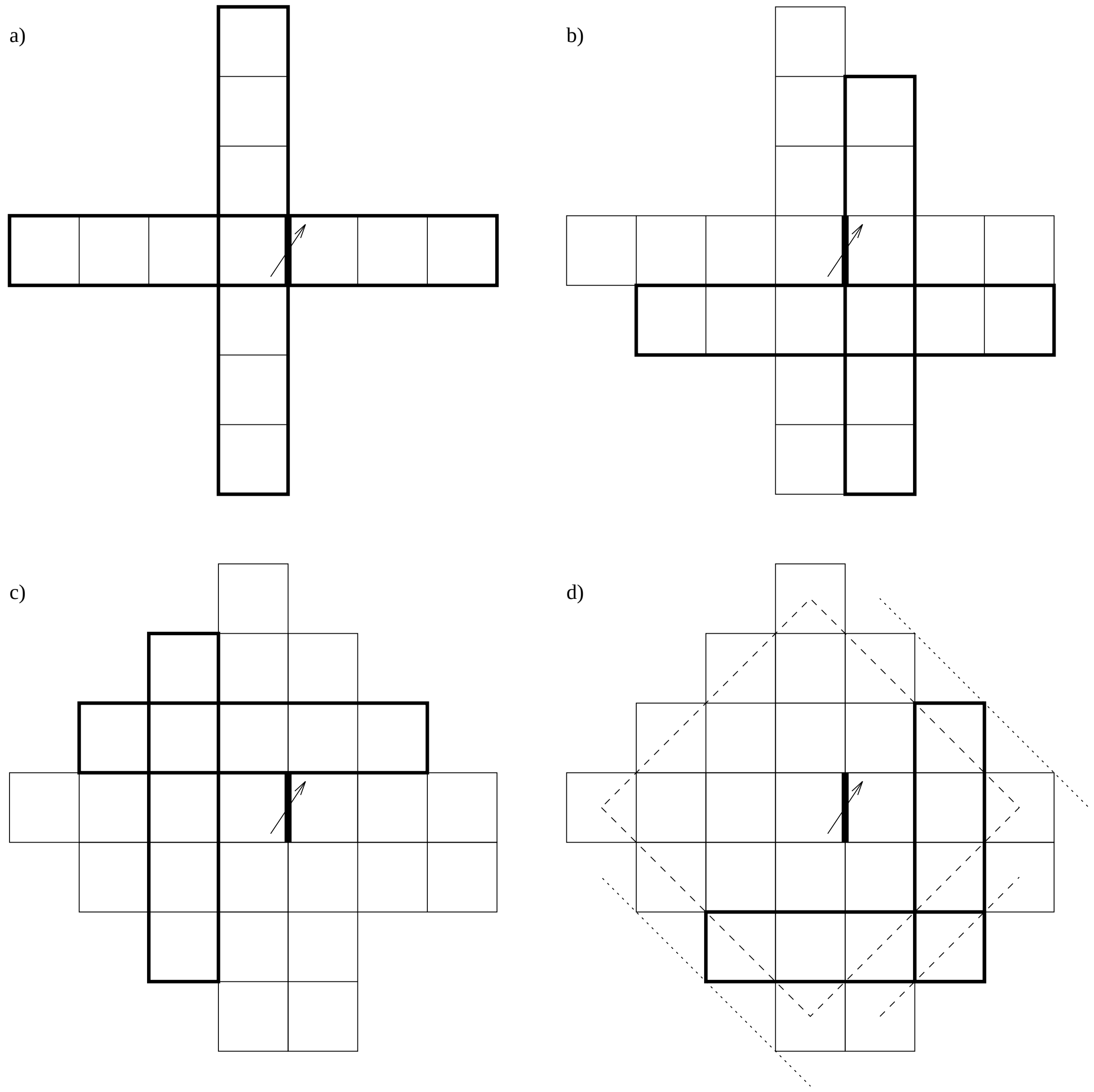
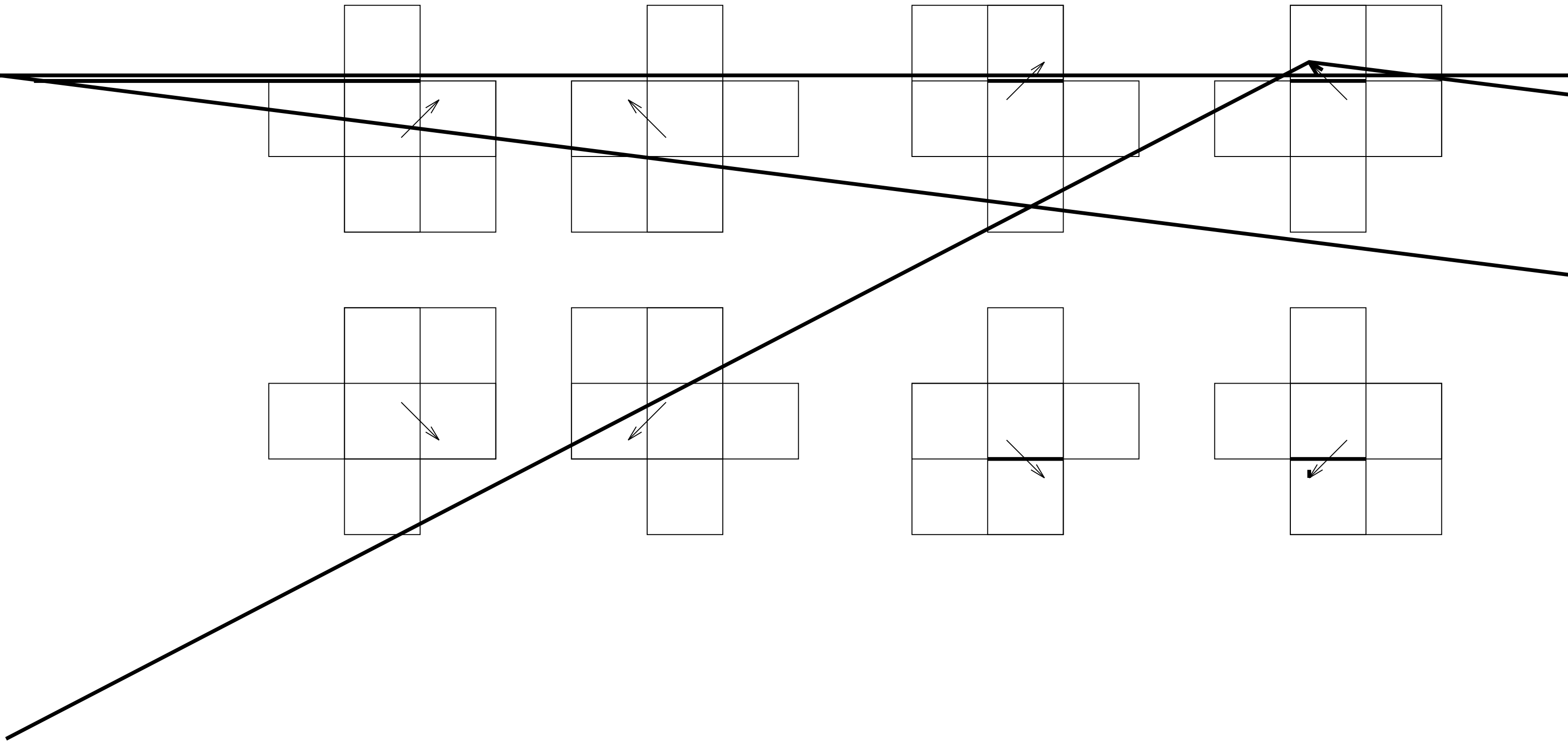


Figure 2.4: Building a stencil for interpolating a sixth order polynomial. a) Seven cells in one row and one column are included for the terms that are powers of x or y. b) Another row and column each of six cells is added for terms xy^n and $x^n y$ where $1 \leq n \leq N-1$. c) Cells are added to ensure there are five cells in another row and column. d) The final cell is added to make a row and column of four cells. A similar method using a seventh order polynomial would result in the cells along the narrow dotted lines also being included in the stencil.

form,

$$\left(\begin{array}{c|c|c|c}
1 & (\Delta x)^m & (\Delta y)^n & (\Delta x)^k(\Delta y)^l \\ \hline
0 & & & \\
\vdots & N \times N & \emptyset & \\
\vdots & & & \\ \cline{2-3}
\vdots & & & \emptyset \\
\vdots & \emptyset & N \times N & \\
\vdots & & & \\ \hline
\vdots & & & \\
\vdots & & & \\
\vdots & \multicolumn{2}{c|}{\emptyset} & K - 2N - 1 \times K - 2 \\
\vdots & & & \times K \\
0 & & &
\end{array}\right.$$

overlap as both fit the constant term) , two rows of at least N cells for the terms $xy,\ x^2y,\dots,\ x^{N-1}y$, two columns of at least N cells for the terms $xy,\ xy^2,\dots,\ xy^{N-1}$, three rows of at least $N-1$ cells for the terms $x^2y^2,\ x^3y^2,\dots,\ x^{N-2}y^2$, three columns of at least $N-1$ cells for the terms $x^2y^2,\ x^2y^3,\dots,\ x^2y^{N-2}$, etc. A third way of building this stencil is N



OOOOOO

of the polynomial being found.

Once we have fitted a polynomial over the region around an edge we can then integrate it over the area swept across the edge to approximate the flux. For the purpose of the following derivation we shall continue to use powers of x and y for the interpolated polynomial which can either be calculated from the Legendre polynomials or used directly. The flux in the x-direction is therefore given by

$$FLUX_x = \frac{1}{u\Delta t\Delta y}$$

strictly a Courant number for this edge. If $\alpha_x = 0$ then this equation will not hold when $k = 0$, but when $\alpha_x = 0$, u must be zero so there will be no flux across the edge. When $\alpha_y = 0$ then v, and therefore w, must be zero, in this case integrating equation (2.29_x) gives

$$FLUX_x = \sum_{n=1}^{K} \phi_n \sum_{k=0}^{N} \sum_{l=0}^{k} \frac{\mathsf{B}_{zn}^{-1}(-u\Delta t)^{k-l}(-\Delta y)^{l}}{}$$

other methods. Most standard methods can be written in the form of equation (2.34) for example, the UTOPIA scheme of Leonard can be written in this form for stencil 1 in Figure 2.5, with

$$\mathsf{G}_{UTOPIA} = \begin{pmatrix} \frac{5}{6} & \frac{1}{2} & \frac{1}{4} & -\frac{1}{3} & -\frac{1}{4} & -\frac{1}{2} \\ 0 & 0 & -\frac{1}{4} & 0 & 0 & \frac{1}{4} \\ -\frac{1}{6} & 0 & 0 & \frac{1}{6} & 0 & 0 \\ \frac{1}{3} & -\frac{1}{2} & -\frac{1}{4} & \frac{1}{6} & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} \\ 0 & 0 & \frac{1}{4} & 0 & -\frac{1}{4} & 0 \end{pmatrix}, \ \boldsymbol{\alpha} = \left(\begin{matrix} 1 \\ \\ \\ \\ \\ \\ \end{matrix} \right.$$

which is related to G_{UTOPIA} by

$$\mathsf{G}_{UTOPIA} = \mathsf{G}\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{3}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{6}{4} \end{pmatrix} . \tag{2.35}$$

The UTOPIA method can be thought of as using a second order polynomial fitted through grid points (Leonard et al. [23]) whilst the new method uses fitting over control volumes. This would cause a slight difference between the coefficients of each scheme and could account for the difference between the two matrices G for each method.

Once the fluxes have been calculated at each edge it is then possible to update the ϕ-value in each cell by simply adding or subtracting the appropriate fluxes,

$$\phi_{ij}^{n+1} = \phi_{ij}^{n} + \sum_{\text{inflow}} FLUX - \sum_{\text{outflow}} FLUX \ . \tag{2.36}$$

Before undertaking this step we may apply a limiter to these fluxes, in order to ensure shape preservation of the ϕ-field.

This method does not impose a time step restriction on the scheme, any value of Δt can be used and the scheme will work. This does not guarantee stability of the generated scheme but we might reasonably expect the stability conditions of the Lax-Wendroff and UTOPIA schemes to hold for our comparable schemes. For higher order schemes we may be concerned about taking small time steps and using parts of high amplitude, high frequency oscillations to calculate our fluxes. If we apply a suitable flux limiter then this will ensure the stability of the schemes possibly by imposing a stability condition on the Courant number. We shall now take a closer look at such a limiter.

2.3.2 The Flux Limiter

We have a simple form for calculating fluxes that requires a matrix-vector multiplic

the following work the density remains constant so this ratio is equal to one, where this is not the case then the Courant numbers will need scaling accordingly. For the limiter to ensure shape preservation it also requires that the sum of the Courant numbers over the outflow edges (the denominators in equations (2.42_{min}) and (2.42_{max})) is less than one. This imposes a time step restriction on the scheme which may or may not be stricter than the restriction on the basic scheme itself. The adjusted fluxes can now be used in equation (2.36) to update the ϕ-field.

2.3.3 Error Analysis

We can use a similar argument to that used in the one dimensional case to try and get an idea of how the numerical error of this scheme. We ag^1

If we adopt the same local origin as when we fit the polynomial then the integrals in the exact case are given by

$$\int\!\!\int_{\mathrm{area(t_n)}} \mathrm{d}y\mathrm{d}x \equiv \int_{-\Delta x}^{0} \int_{x-\int_{n\Delta t}^{(n+1)\Delta t} u(t,x,0)\ \mathrm{d}t\ \ +0-\int_{n\Delta t}^{(n+1)\Delta t} v(t,x,0)\ \mathrm{d}t}^{0} \mathrm{d}y\mathrm{d}x\ , \tag{2.48$_x$}$$

and $$\int\!\!\int_{\mathrm{area(t_n)}} \mathrm{d}x\mathrm{d}y \equiv \int_{-\Delta y}^{0} \int_{0-\int_{n\Delta t}^{(n+1)\Delta t} u(t,0,y)\ \mathrm{d}t\ \ +y-\int_{n\Delta t}^{(n+1)\Delta t} v(t,0,y)\ \mathrm{d}t}^{0} \mathrm{d}x\mathrm{d}y\ , \tag{2.48$_y$}$$

and for the approximate case by

$$\int\!\!\int_{\mathrm{area_n}} \mathrm{d}y\mathrm{d}x \equiv \int_{-\Delta x}^{0} \int_{x-\int_{n\Delta t}^{(n+1)\Delta t} u_{ij}^{n}\ \mathrm{d}t\ \ +0-\int_{n\Delta t}^{(n+1)\Delta t} v_{ij}^{n}\ \mathrm{d}t}^{0} \mathrm{d}y\mathrm{d}x$$

then there will also be a contribution to the error from the approximation of the area swept across the edge. If the limiter is applied to the fluxes then the above argument will not hold when the limiter changes the value of a flux.

Summing over each cell to find a bound on the global error gives,

$$\left|e^{n+1}\right| \leq |e^{n}| + 2\sum_{i}$$

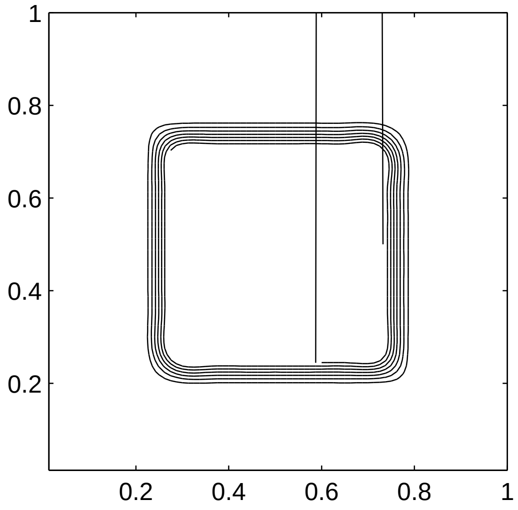
The tests in themselves do not give a complete picture of how well the scheme would perform in the atmospheric problem. The uniform flow, Section 2.4.1, has little resemblance to flows found in the real atmosphere but does provide an indication of how well the scheme handles flow at an angle to the grid. The rotational flow, Section 2.4.2, has more in common with rotating systems in the atmosphere but the flow has a zero strain rate and so does not stretch parcels in the same way. The final test, Section 2.4.3 on deformational flow does include vortices which stretch fluid parcels but these are regular and do not interact with one another as in the real atmosphere. Looking at the results of these tests together should give a good idea of the strengths and weaknesses of the scheme and where it will succeed or have problems in the real atmosphere.

For completeness, several measurements were taken of errors and properties of the advected profile. For the fi of the distribution and of how well the scheme maintained the symmetry of solutions (where it should). The maximum and minimum values were used primarily to check that the limiter is working and not allowing the values of the extrema to grow, as was the case in all runs. For symmetric initial conditions in a symmetric flow the solution should also be symmetric, and is to machine accuracy for these test cases. A qualitative account of the results for the third test case is presented here.

to northeast) and a Courant number of one quarter ($\alpha_x = \alpha_y = 0.25$) was used in each direction. This meant that each doubling of the resolution requires twice the number of time steps to be used. The profile is advected once around the periodic domain in one second, requiring 80, 160, 320 or 640 time steps depending on the grid size.

Two initial profiles were used for this test case, a square step and a smooth $\cos^2$ function. The tracer value for the square step initial conditions was one in the region $0.25 < x, y \leq 0.75$ and zero elsewhere. The smooth function was defined as $\cos^2(2\pi R)$ for $R < 0.25$ and zero elsewhere, where R is the distance from the centre of the domain. The step initial conditions test how well the schemes can maintain a sharp gradient in a smooth velocity field and should show how the limiter affects the accuracy in preventing spurious oscillations in the solution. The smooth initial condition tests how well the schemes are able to maintain a localised extreme value.

The L_2 and L_∞ errors are used to show the relative accuracy of the schemes. The former gives a measure of the accuracy over the whole domain whilst the latter gives the largest error in any cell. The L_2 e initwhill


1
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
1

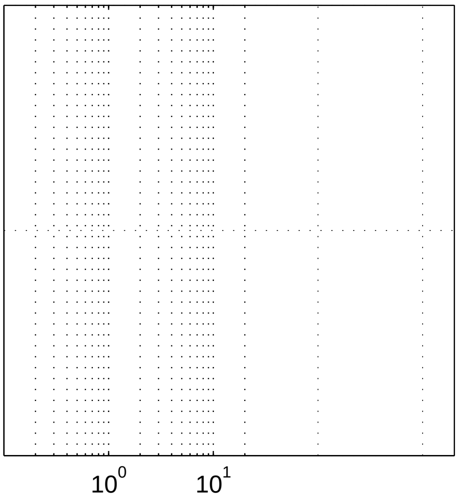
limiter applied, then these oscillations are removed, as seen in plot b. The downside of this is that there is a little more smoothing of the sharp edges.

If a fourth order polynomial is used instead of the second order one, then we get the result shown in plot c. This solution contains more oscillations but they are smaller in both magnitude and wavelength, so large gradients are better maintained. If a higher resolution grid is used instead of a higher order polynomial then the results are very similar, as seen by comparing plots c and d. The main differences are that the oscillations are a little smaller on the finer grid and that this result took more than twice as long to compute (as will be seen in plot a of Figure 2.8)

The final solutions of the same schemes applied to the smooth initial condition are not shown but do not exhibit any significant differences in the behaviour of the schemes. Small oscillations are still generated around the non-zero part of the initial profile, though they are smaller in magnitude. The main difference in results between the two profiles is the maintenance of the peak value in the smooth initial profile. This difference is best seen in the later comparison between the L_∞ errors of the different schemes.

Figure 2.8 shows the L_2 and L_∞ norms plotted against the computational cost for both profiles, with and without the use of the flux limiter. This plot and ones like it will be used widely throughout the next few chapters so it is important to take some time to understand what they show. Each symbol on the plot represents one of the error measures for one of the schemes, plotted according to the error value and the computational cost (CPU runtime) taken to calculate the result.

The symbols are joined by lines according t


10^0
10^1

on which the scheme was used and the type of error being measured. Solid lines represent either the L_2 error or the error due to di haviour of the different order schemes on di oscillations and improves the phase speed, which is shown by the larger improvement in accuracy in plot a. Applying the flux limiter removes these oscillations and improves the accuracy, in the L_2 norm, for low order schemes. For the higher order schemes, these oscillations are smaller and errors due to the poorer representation of the sharp gradient begin to dominate. The effect of this is that the errors appear to be approaching a limit as the order of the polynomial is increased. These properties of the different order schemes appear to be the same here as in the one-dimensional case.

The results for the smooth initial profile, plots c and d, display slightly different behaviour, due to the absence of a sharp gradient in the initial profile. The errors are much smaller than the previous case, which is a result of the smooth profile being more amenable to accurate interpolation by polynomials. The L_∞ error for this case is effectively a measure of the change in the peak value of the solution and this improves with increasing order and resolution. The L_2 error also shows a continued reduction as higher order polynomials and higher resolutions are used. The relationships between even and odd order schemes observed in the previous case are again present in this one.

When the limiter is applied, the effect is similar to the previous case, in that there is a slight improvement in the errors for some of the lower order schemes and a worsening for the higher order schemes. The main difference in this case is that there are oscillations between the errors of high-even and high-odd order schemes. High-even order schemes appear to be approaching a limit form below whilst high-odd order schemes would seem to be approaching the same limit from above. One possible mechanism for this is the difference between the number and size of oscillations in the even and odd order schemes. The limiter will alter the fluxes more where many, large oscillations are being generated and this will have a greater effect on the accuracy.

The x-axis on these plots indicates the computational cost (CPU runtime) for generating and running the schemes. This allows us to make a fairer comparison between the schemes as it shows the computational effort required to reach the accuracy achieved by each scheme. In the un-limited cases, plots a and c show that a given accuracy in the L_2 error can be reached by either increasing the order of the interpolating polynomial or the resolution. In these cases however, the CPU runtime required to gain a given level of accuracy is roughly the same, whichever method of improvement is used. For example, if we want the L_2 error to be less than 0.08 in plot a of Figure 2.8 we can achieve this in around three and a half CPU seconds using either a sixth order scheme on a 20×20 grid or a second order scheme on a 40×40 grid. Alternatively, if we want the most accurate solution possible in forty CPU seconds for the smooth initial profile, an eighth order scheme on a 40×40 grid gives a similar L_2 error to a third order scheme on an 80×80 grid.

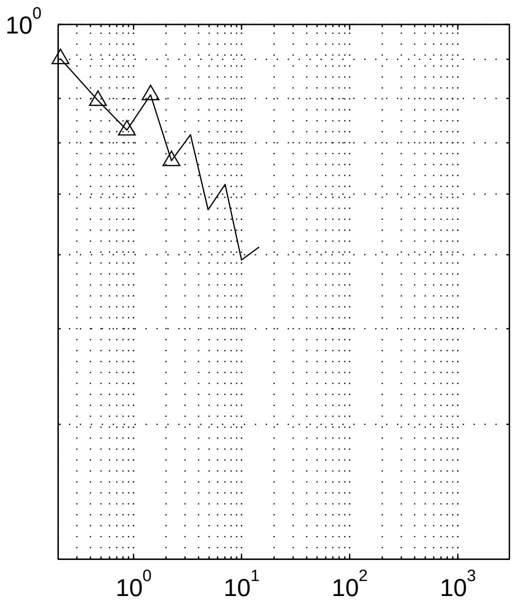
When the limiter is used all the schemes become more expensive to run and the errors are generally worse than in the unlimited cases. This is the trade offs for resolutions between those shown in plot b.

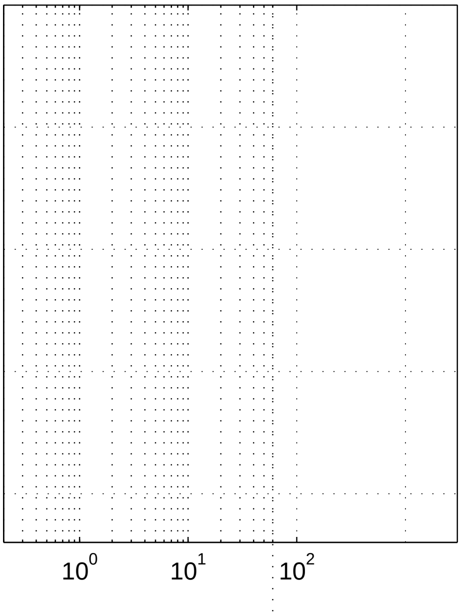
In order to try and better understand why the errors behave in this way, we can look at the diffusive and dispersive components of the L_2 norm. The square of the L_2 error measure may be written in terms of diffusion and dispersion errors, derived through writing the L_2 error in terms of mean and variance. Following the argument of Takacs [49],

$$L_2^2 = \frac{\sum(\phi_T - \phi_N)^2}{\sum \phi_T^2} = \frac{1}{\sum \phi_T^2} \big(\sigma^2(\phi_T - \phi_N) + (\bar{\phi}_T - \bar{\phi}_N)$$

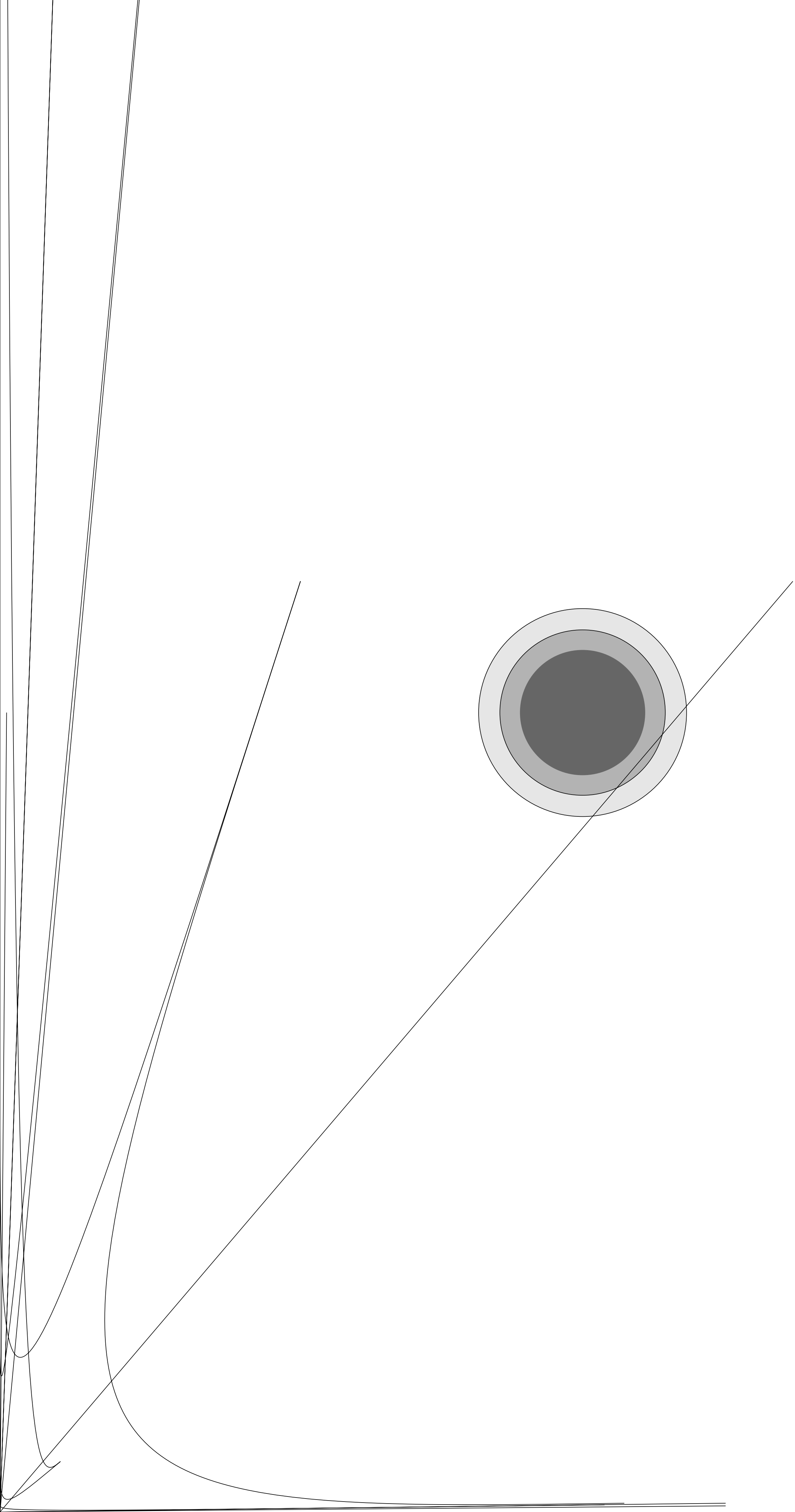
L

clearly from these plots. Increasing from an odd to even order polynomial introduces more diffusion but reduces the dispersion. Increasing from an even to an odd order polynomial has little effect on the dispersion but does reduce the amount of diffusion. In the case of the smooth p_m

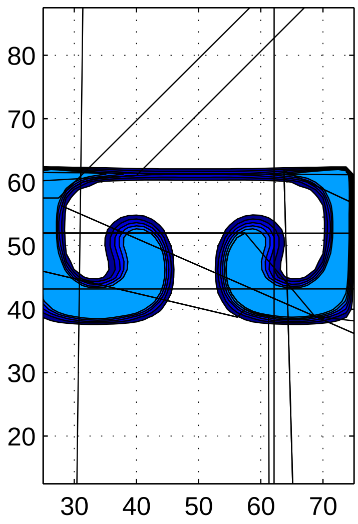

10^0
10^0
10^1
10^2
10^3


10^0
10^1
10^2

The diffus



vortices



80
70
60
50
40
30
20
30
40
50
60
70

alent to five rotatio

over the lower order schemes on a given grid because they are better at maintaining the steep gradients involved for longer and thus resolving the features closer to the grid scale. The one advantage that the lower resolution schemes have is that they are much cheaper to compute. Doubling the resolution means that calculations must be performed for four times as many cells at each time step and that twice the number of time steps need to be taken in order to satisfy any Courant number condition. This means that doubling the resolution causes a roughly eight fold increase in the computational cost of the run. Whether or not this increased computational burden is worth the effort depends on how important it is that the scheme resolves the small scale features.

2.5 Summary

A general method for generating multidimensional advection schemes of improving accuracy on a regular rectangular grid has been developed and implemented in two dimensions. The method works by fitting a polynomial to the discrete tracer field around an edge and then integrating this polynomial over the region swept across that edge to find the flue t ce . T

error, which is also shown in these results. I , a similar method on a non-rectangular grid then we will need to find a different region over which to fit the interpolating polynomial around an edge. It is this problem that we now consider in the next chapter, along with a change to the way that the polynomial is fitted over the stencil.

C

whic

a fluid of constant density,

$$\frac{\partial \phi}{\partial t} = -\frac{\partial (u\phi)}{\partial x}. \tag{3.1}$$

Integrating this over a time step Δt and a cell in space we get the exact formula for the update equation,

$$\phi_i^{n+1} = \phi_i^n - \left(\hat{\phi}_{\mathcal{R}}^n - \hat{\phi}_{\mathcal{L}}^n\right) \ , \tag{3.2}$$

where ϕ_i^n is again the mixing ratio of tracer ϕ in the i^{th} cell at the n^{th} time step and $\hat{\phi}_{\mathcal{L},\mathcal{R}}^n$ is the flux across the left/right edge of cell i in time Δt. For a fluid of uniform, constant density ϕ can also be thought of as an average concentration. Discretising this equation as before gives

$$\phi_i^{n+1} = \phi_i^n + \left(\mathbf{c}_{\mathcal{L}} \cdot \boldsymbol{\phi}_{\mathcal{L}}^n - \mathbf{c}_{\mathcal{R}} \cdot \boldsymbol{\phi}_{\mathcal{R}}^n\right) \ , \tag{3.3}$$

where the $\boldsymbol{\phi}_{\mathcal{L},\ \mathcal{R}}^n$'s are vectors of the mixing ratio of the tracer in the cells surrounding the left/right edges and the $\mathbf{c}$'s are the coefficients that our method will calculate. In order to calculate the $\mathbf{c}$'s when we do not have the same number of cells in our stencil as there are terms in the polynomial we shall insist that the residual, r, is minimised where

$$r^2 = \sum_{\text{stencil}} \left(\Delta x_i \phi_i^n - \int_{cell\ i} p_N(x)\ \mathrm{d}x\right)^2 \ . \tag{3.4}$$

This minimises, in a least squares sense, the difference between the previously predicted amount of tracer in each cell and the amount approximated by the fitted polynomial.

In the previous chapter we forced $r = 0$ by insisting that the tracer amount approximated by the polynomial was equal to the predicted amount in each cell in the stencil, or

$$\Delta x_i \phi_i^n = \int_{cell\ i} p_N(x)\ \mathrm{d}x. \tag{3.5}$$

steppin

fi

re-arranging gives

$$a_N = \frac{\left(\sum_{j=0}^{N-1} \acute{a}_j\right)\left(\sum_{j=0}^{N-1} \frac{\Delta x^N}{\Delta x^{j-1}} \mathsf{B}\ {}^{\mathrm{b}}_{A} \left(\acute{\mathsf{B}}_j^{-1} \cdot \mathbf{b}\right)\right)}{1 + \left(\sum_{j=0}^{N-1} \ \Delta x^N\right.}$$

a least squares fit in this case does not result in the integral of the polynomial over each cell being equal to the amount of the advected quantity in that cell. Such a scheme would therefore not give the exact solution if a Courant number of one is used, as many one dimensional numerical schemes do.

In order to test this method in one dimension we shall try fitting an N^{th} order polynomial over $N+2$ cells using the method described in Section 3.2. When we do this we find that growing oscillations are generated upwind of discontinuities in the gradient of the advected profile. These oscillations grow much more slowly than those for the under-determined case and are only generated upwind of discontinuities. If these problems are simply caused by no longer insisting on equality bee a eu ica it a

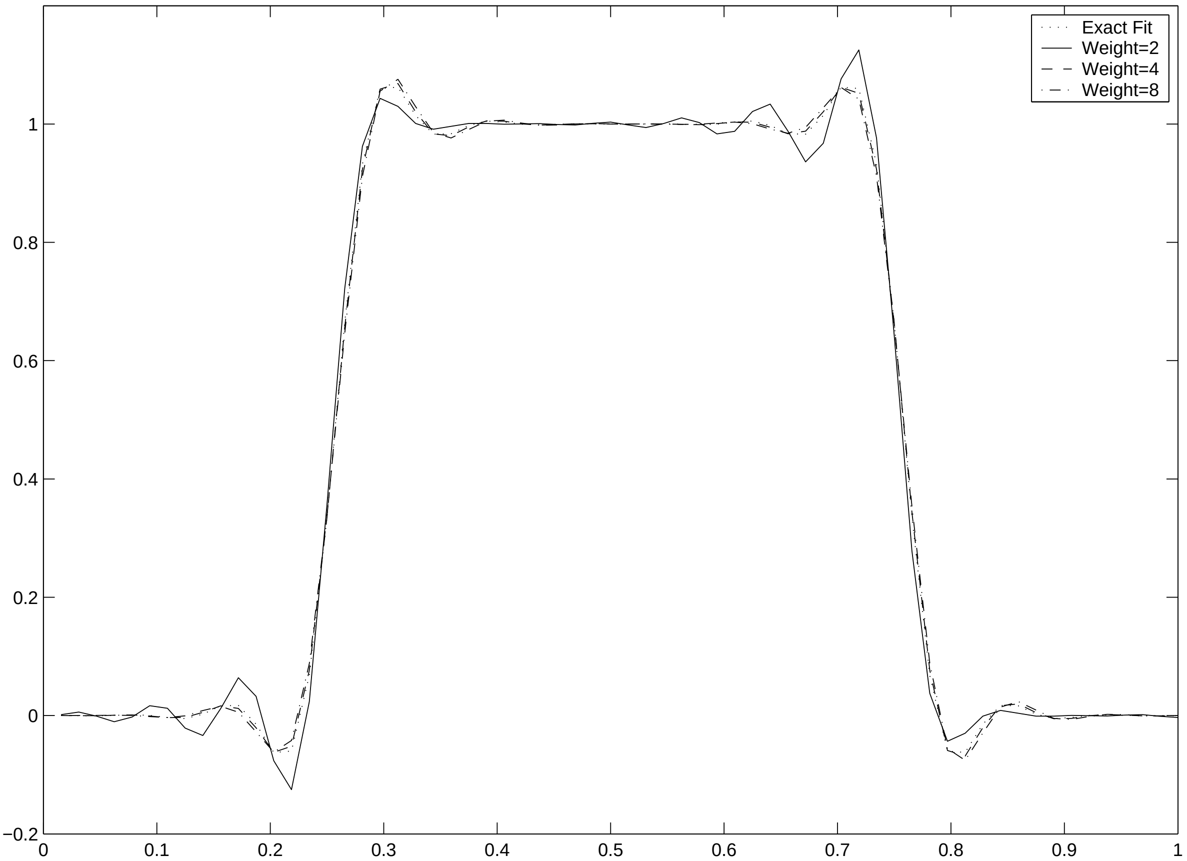
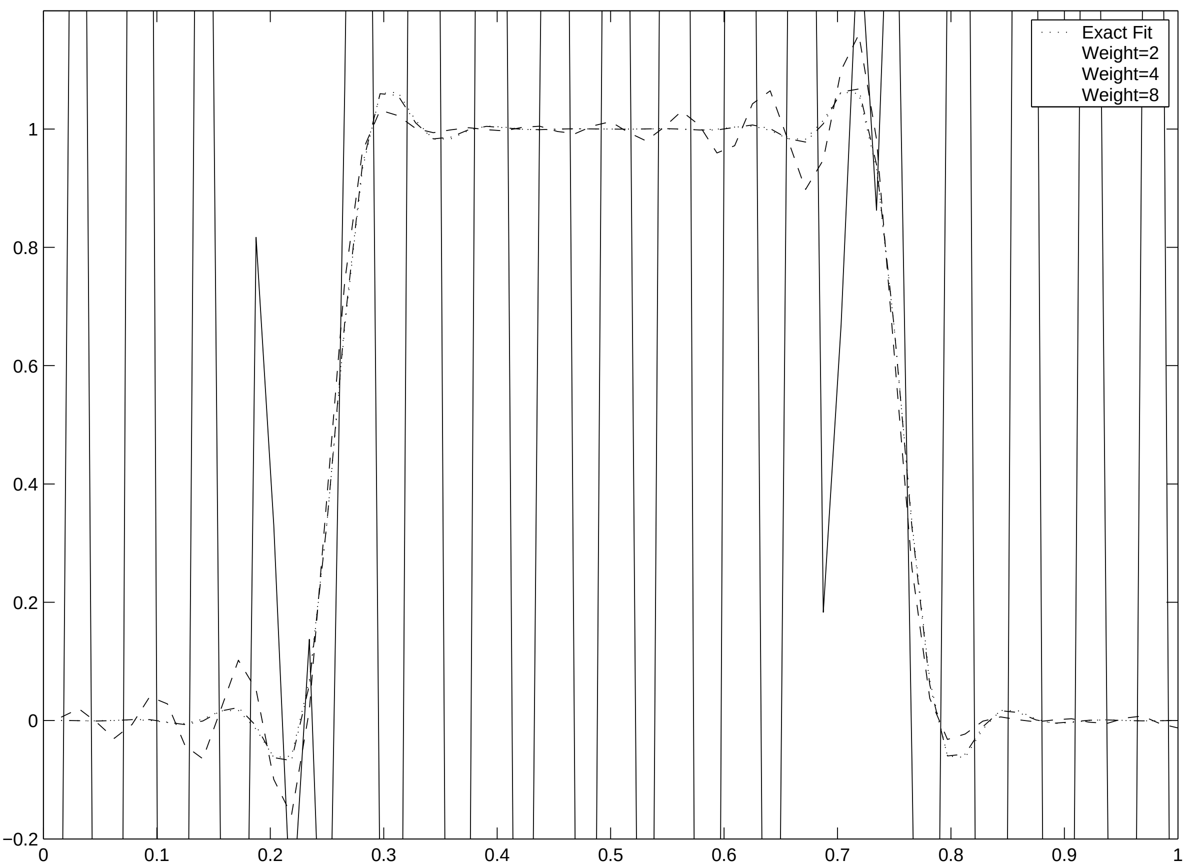


Figure 3.1: Num_m


Exact Fit
Weight=2
Weight=4
Weight=8
1
0.8
0.6
0.4
0.2
0
−0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

on other grids in order to decide on suitable values to use for the weights. Applying these ideas in two dimensions will enable us to tackle problems on two dimensional grids where we need to select stencils that contain more cells than there are terms in the polynomial we are fitting. Before attempting this we shall see what a similar error analysis to that used in the previous chapter can tell us about the accuracy of these schemes.

3.2.3 Error Analysis

Following the same argument that was used in the previous chapter, we define the error in a cell at time t_n to be the difference between the actual amount of the tracer in that cell and the approximated amount

$$e_i^{n+1} = \phi_i^{n+1} - \int_{(i-1)\Delta x}^{i\Delta x} \phi(t_{n+1}, x)\mathrm{d}x \ . \tag{3.21}$$

The right hand side can be written in terms of the values at a previous time and the fluxes across each edge of the cell to give

$$e_i^{n+1} = e_i^n + \sum_{\text{edges}} \left(\int_{\mathrm{area_n}} p(x)\mathrm{d}x - \int_{\mathrm{area(t_n)}} \phi(t_n, x)\mathrm{d}x \right) \ . \tag{3.22}$$

If the continuous and discrete wind fields are the same in this region for this time then

$$e_i^{n+1} = e_i^n + \sum_{\text{edges}} \int_{-u_0\Delta t}^{0} \left\{ p(x) - \phi(t_n, x) \right\} \mathrm{d}x \ . \tag{3.23}$$

Summing over each cell to find a bound on the global error gives,

$$\left| e^{n+1} \right| \leq |e^n| + 2\sum_i \int_{i\Delta x - u_0\Delta t}^{i\Delta x} \left| p_{i_N}(x) - \phi(t_n, x) \right| \mathrm{d}x \ , \tag{3.24}$$

where i runs over all edges on the grid and $p_{i_N}(x)$ is the N

we do not have as much information about the accuracy of this interpolation as before. We would expect that the accuracy of fitting th

3.3.1 Generating the Advection Scheme

We begin with the two dimensional advection equation in flux form,

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) = 0 \ , \tag{3.25}$$

where ϕ and $\mathbf{u}$ are two dimensional tracer and wind fields respectively. Discretising the time derivative gives

$$\phi_i^{n+1} = \phi_i^n - \sum_{\text{edges}} \hat{\phi} \ , \tag{3.26}$$

with $\hat{\phi}$ representing the flux across an edge of cell i in one time step. We wish to approximate these fluxes by the integral of a polynomial over an approximation of the region swept across the edge at each time step,

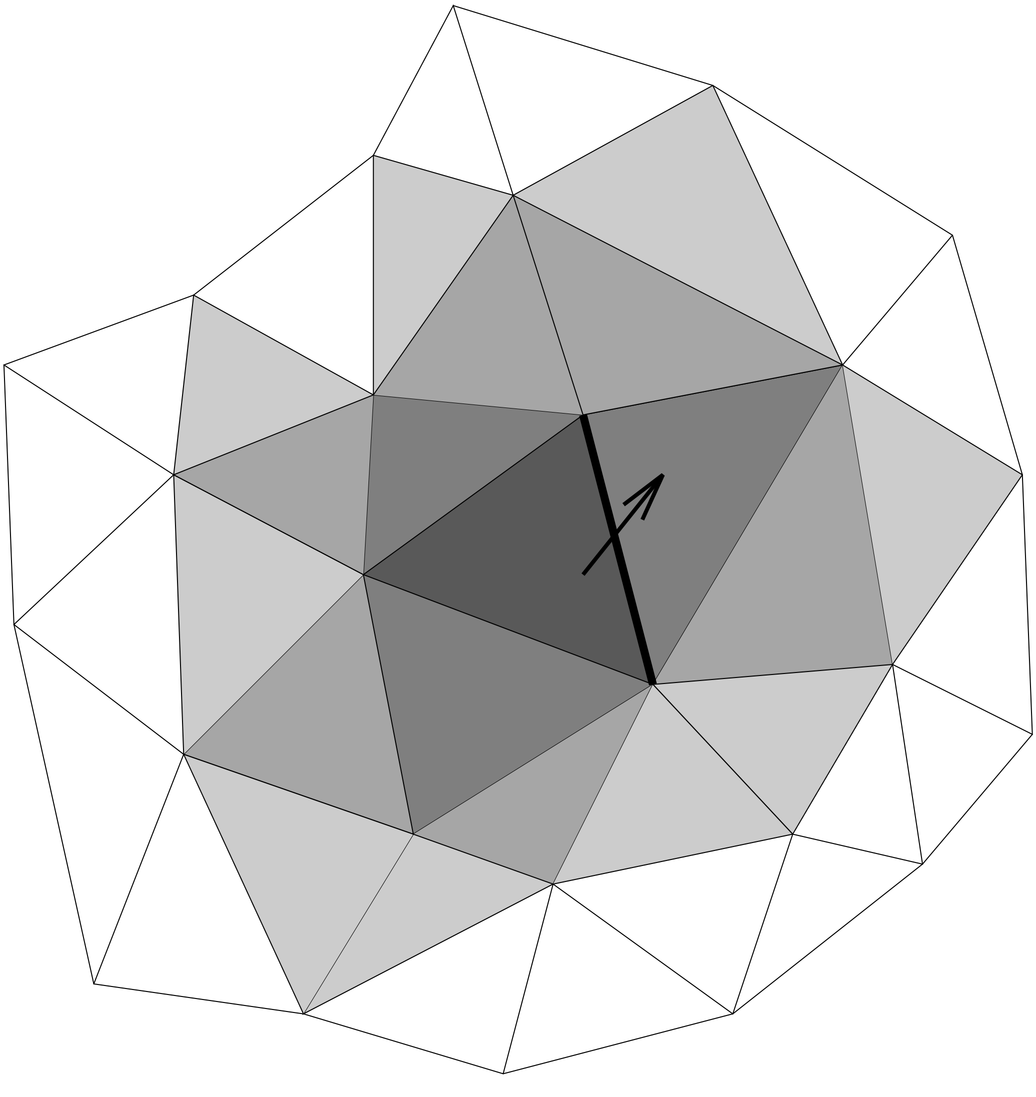
$$\hat{\phi} = \frac{\alpha}{\ell \, |\mathbf{n}| \, \Delta t} \int_{|\mathbf{n}|\Delta t} \int_{\ell} p_{_N}(x, y) \mathrm{d}x \mathrm{d}y \ , \tag{3.27}$$

where $\alpha = \frac{\ell |\mathbf{n}| \Delta t}{A}$, ℓ is the length of the edge, A is the area of cell i and $|\mathbf{n}|$ is the magnitude of the flow normal to the edge. The origin of the local $x - y$ coordinate system is defined separately for each cell. The polynomial $p_{_N}$ is found by fitting over a stencil such that

$$\int \int_{\text{cell i}} \sum_{k=0}^{N} \sum_{l=0}^{k} a_z x^{k-l} y^l \mathrm{d}x \mathrm{d}y = A \phi_i^n \ , \tag{3.28}$$

for each cell i in the stencil, where again $z = (k+1)k/2+l$. The difference between this method and that of the square grid now arises since we must find a different way of choosing the stencils over which to fit the polynomial.

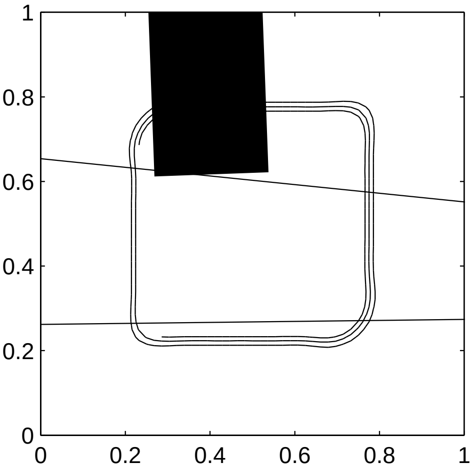
On a general triangular grid the cells may not be aligned with one another so we will not be able to find a simple stencil as in the last chapter. The first order upwind scheme will use the cell upwind of an edge so this would be a suitable cell to begin our stencil. If we fit a zeroth order polynomial over this cell we will generate a first order upwind scheme. If we wish to use a first order polynomial then we will need to add two or more cells neighbouring this central cell, since there are three terms in the polynomial. On an irregular grid, distances between the centres of cells or their size could be used to select two cells to add. On a regular grid however, there may be no way to choose between cells other than using the flow direction. Using the flow to choose which ce

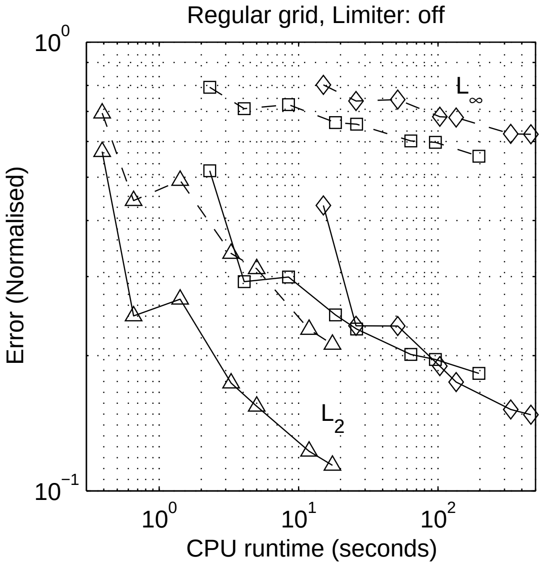


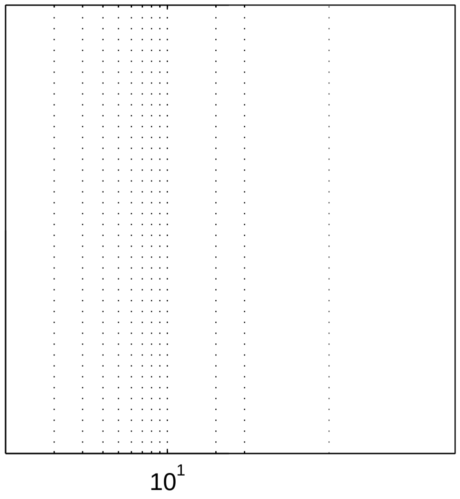
used whenever it is upwind of an edge.

The need for this change could be eliminated altogether by beginning the stencil with the two cells either side of the edge and adding their neighbour

in the upstream neighbourhood of I1 if there is flow from these cells into A. If B is in the upstream region then we must also include D if there is flow from this cell to B.


1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
1

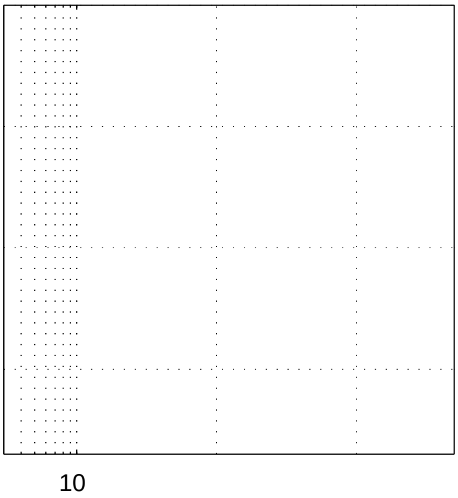

Regular grid, Limiter: off
10^0
10^{-1}
Error (Normalised)
L_∞
L_2
10^0
10^1
10^2
CPU runtime (seconds)


10^1

condition as well as other cases.

In all other tests on this and the square grid, the initial step region was aligned with the grid cells and was thus more accurately resolved. For this resolution grid the gradient falls in the middle of cells and so is effectively smoothed in the initial condition. It is this initial condition that forms the fi

u


10

sition applied to the same cases as above. In all cases the zeroth order scheme is shown to be highly diffusive but the higher order schemes have a dispersion error significantly larger than the diffusion. The dispersion error decreases with increasing order but more slowly than the diffusion error. This suggests that the higher order schemes maintain the steep gradients more effectively but make little improvement in the phase error.

Although these results on the triangular grids are not as good in quantitative terms as those on the square grid, they are qualitatively very similar. The behaviour of the schemes as the order is increased and the greater effect of the limiter on higher order schemes are particularly significant features present in both cases. The results for the advection of a smooth profile by the same flow as the square step behave in a similar way to the results above and on the square grid. They are not shown here but they are also quantitatively worse than the square grid cases whilst being qualitatively very similar.

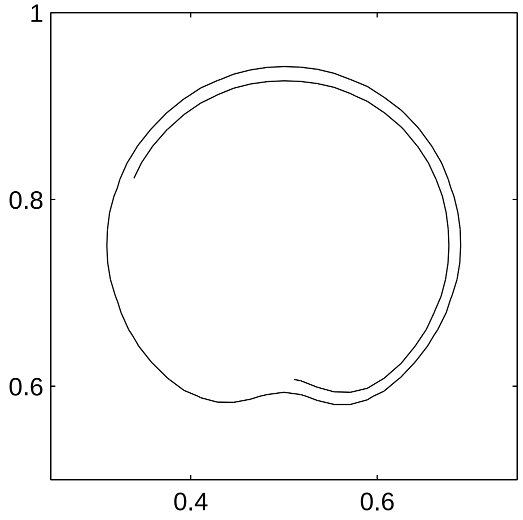
The computational cost of generating and running these schemes on the triangular grids is two or three times larger than that of the square grid tests. This is mainly due to the number of stencils that are being used in each case. On the square grid there were only seven different stencils over which a polynomial needed fitting, on the triangular grid there was one stencil for each grid cell. The computational cost of running the schemes, ignoring the cost of their generation, depends only on the number of edges and the number of cells in each stencil. There are roughly $\frac{3}{4}$ the number of edges on the triangular grid compared to a similar resolution square grid but there will be a few more cells in the stencil of a method of comparable order.

3.4.2 Rotational Flow

For this test case an anti-clockwise solid body rotation was applied to an initial profile of a split cylinder in the same way as in Section 2.4.2. The rotation speed was 2π times the distance from the centre of the rotation so the solution after one second was the initial distribution. The same grids as used in the previous section were again used for this test. The time step was chosen to give a maximum Courant number of 0.5 across any edge of the grid. The final numerical solutions for the region around the cylinder from four of the schemes are shown in Figure 3.10

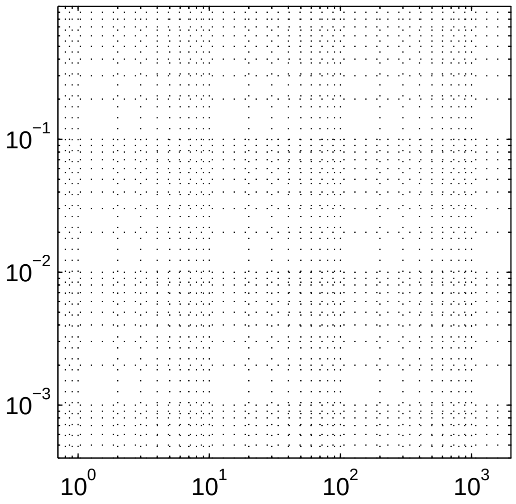
The first of these plots shows the result of using a second order scheme on a regular grid without the limiter. The slot shows a significant amount of filling and there is some erosion of the bridge. This scheme also displays the usual spurious oscillations and negative values that we have seen in other cases. Plot b shows the resulting solution when the flux limiter is applied to the fluxes generated by this scheme. This has removed the spurious oscillations and negative values but has caused more filling of the slot and erosion of the bridge. The solution does not show any of the anisotropic distortion that was seen in uniform flow case, most likely because the flow is rotational and there is no single direction in which the profile is stretched. Increasing the order of the scheme to four gives the result shown in plot c. As before, the steep gradients are better maintained and there are more, smaller oscillations. This leads to less filling of the slot and better maintenance of the bridge. Using the fourth order scheme on the irregular grid gives a solution very similar to the the one on the regular grid.

Figure 3.11 shows the L_2 and L_∞ norms for all cases on the regular


1
0.8
0.6
0.4
0.6

and irregular grids run with and without the limiter. On the regular grid without using the limiter the L_2 error behaves as in previous cases, decreasing with increases in the scheme order and grid resolution. On the coarsest grid the L_∞ error and L_2 error are very similar, a result of severe diffusion of the advected profile which loses almost all of the split cylinder shape. This is the case for zeroth order schemes on all grids but at higher resolutions the L_∞ errors is significantly larger than the L_2 error. The two error measures do behave in a similar fashion as the order of the schemes and resolutions of the grid are increased.

The results for schemes run without the limiter on the irregular grid (plot c) are very similar to those on the regular grid. The main difference is that the L_∞ errors are larger than on the regular grid, most likely because of the variation in cell sizes. When the limiter is applied to these schemes (plots b and d) the observed trends are barely altered. There is a general increase in the values of these error measures over the un-


10^{-1}
10^{-2}
10^{-3}
10^{0}
10^{1}
10^{2}
10^{3}

Once again these results have a very similar form to those of the square grid and to those of the uniform flow test case. In this case the values of the error measures are a little better than on the square grid. This is again because of the way in which the initial conditions were set up in this case. Integrating the specified initial condition over each cell led to some smoothing of the sharp gradient in the initial condition and thus the 'true' solution.

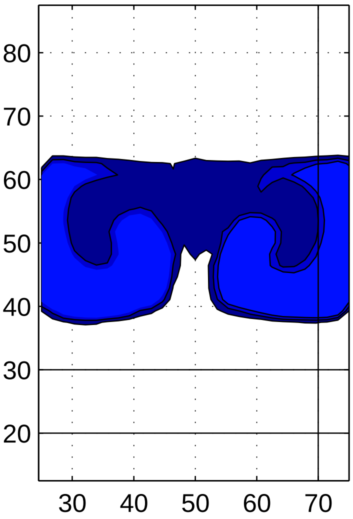
3.4.3 Deformational Flow

This case is set up in exactly the same way as for the square grid test. A flow field of 16 self contained, counter rotating vortices is used to advect the initial profile of a cone of height one unit over a domain 100 units square. The initial condition is only non zero over the six central vortices so the solution should be zero over the remaining vortices for all time. The initial profile is stretched out by the vortices an

The results on the triangular grid are not symmetric in this case. The flow field and the initial conditions are symmetric and so should be the solution. The triangular grid does not share these symmetries and so neither do the solutions. Even with this difference we would expect the results to be more symmetric than they are. The reason for the asymmetry in these solutions is the anisotropic distortion caused by the flux limiter. This is only manifest when the flow is at an angle to the grid and so there have been different amounts of distortion over each vortex.

The effect of this distortion over the longer time period can be seen in Figure 3.14. After one rotation around the vortex we have still managed to capture the features of the flow with a streamer of tracer being wrapped around the vortex. This streamer is broader than it should be however and this can be seen in the profiles at later times. After five rotations around the vortex at $T/10$ we have lost all features of the tracer field. The only feature of the model that is 'correct' is that none of the tracer has crossed between vortices. The 'eggcup' profile is not visible in any of the profiles at time $\frac{T}{10}$ or later.

The results for this test on the irregular grid are very similar to those of the regular grid and are not shown here. On the irregular grid previously described the grid does not align exactly with the boundaries of the vortices. This will allow tracer to 'leak' from the central vortices into neighbouring ones in which they


80
70
60
50
40
30
20
30
40
50
60
70

those of the square grid, there is much more smoothing of the profile in all cases on the triangular grid. This smoothing is mainly caused by the flux limiter on the triangular grid but there may also be a contribution from the size and shape of the grid cells.

3.5 Summary

The method developed in the previous chapter has been adapted for use on triangular grids. This adaption was made by changing the way the polynomial is fitted over the region around each face. This was necessitated by the method of selecting stencils so as not to require the stencils to change whenever the flow changed. This in turn was desirable in order to keep down the computational cost of the scheme.

This method can now be applied to any grid on the plane. It may be possible to find different ways of selecting stencils on other grids as we did on the square grid. This may be of benefit to the schemes but neither this change nor any others should be required for the method to work. The accuracy of the schemes on other grids cannot be assured but the tests run so far show that the behaviour of the schemes varies little on different grids.

The test cases run using this new method of fitting the polynomial showed little difference in the behaviour of the schemes from the previous method as the grid re ut o

Chapter 4

Icosahedral-Hexagonal Grid on the Sphere

4.1 Introduction

T

the sphere we will begin by looking at what effect this has in one dimension by projecting the circle onto a line in Section 4.2. In Section 4.3 we will take a closer look at the icosahedral-hexagonal grid before applying the method to this grid in Section 4.4. The results of some test cases are presented in Section 4.5. The suitability of the method for inclusion in a more complex model is discussed in the final section, Section 4.6, before the method is applied to a shallow water model in the next chapter.

4.2 One Dimensional Advection on a Circle

We have already looked at how to apply this type of method to irregularly spaced grids in one dimension in chapters 2 and 3. In this case the grid will be defined on a circle by a regular angle, $\Delta\theta$, giving cells of uniform arc length, $r\Delta\theta$ where r is the radius of the circle, though this need not be the case. We can select a stencil over which to fit a polynomial in the same way as before. We choose $N+1$ cells centred either on the edge of interest or the cell upwind of it which are projected onto a line before we fit an N^{th} order polynomial over them.

Figure 4.1 shows two ways in which the circle can be projected onto the line. The line has been defined as the line through the two points that form the edges of cell 0. Cell 0 is the central or upwind cell of the stencil so we shall have a different line for each cell. The width of this cell, Δx_0, is found using the cosine rule

$$\Delta x_0 = \sqrt[+]{2r^2(1-\cos\Delta\theta)}\ . \tag{4.1}$$

Δx_1 will have a length that depends on the way in which we project the circle onto the line. If we project along the normal to the circle at the

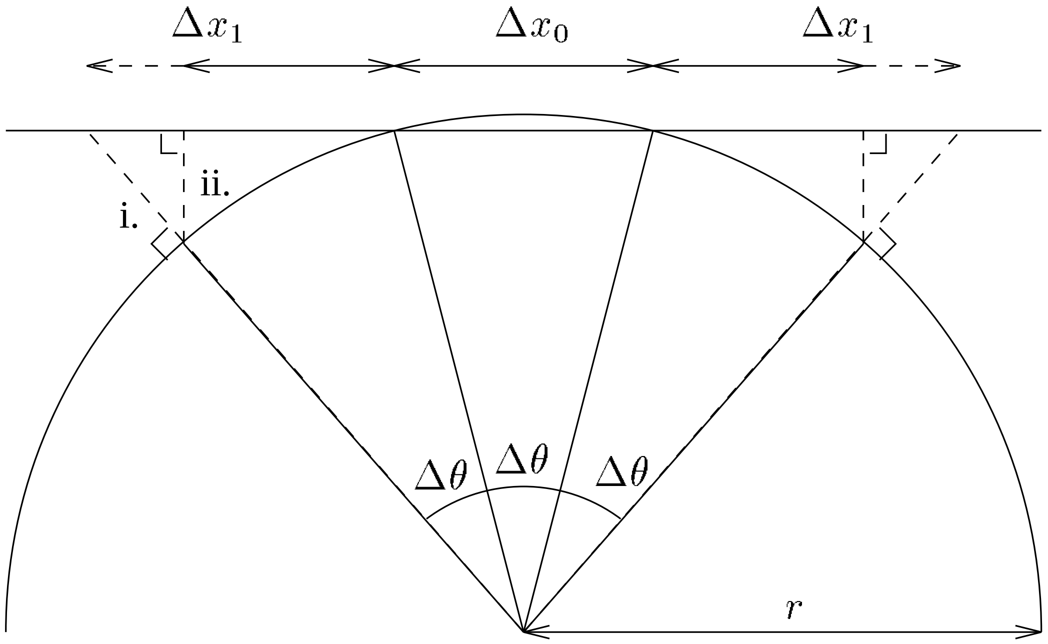


Figure 4.1: Projecting the circle onto (

If we follow the same method as in Section 2.2, using a second order polynomial and a Courant number of one half, then the equation for the flux (2.16) has the form

$$\hat{\phi}^n = \frac{-\left(1-2f(\Delta\theta)\right)\phi_{-1} + \left(4+12f(\Delta\theta)+8f(\Delta\theta)^2\right)\phi_0 + \left(1-2f(\Delta\theta)\right)\phi_1}{4f(\Delta\theta)\left(1+f(\Delta\theta)\right)\left(1+2f(\Delta\theta)\right)}\ , \tag{4.4}$$

where the function $f(\Delta\theta)$ depends on the method of projection. The function f takes the following forms,

$$f(\Delta\theta) = 1 \qquad \text{where } \Delta x_n = r\Delta\theta \quad \text{(circular case)} \tag{4.5}$$

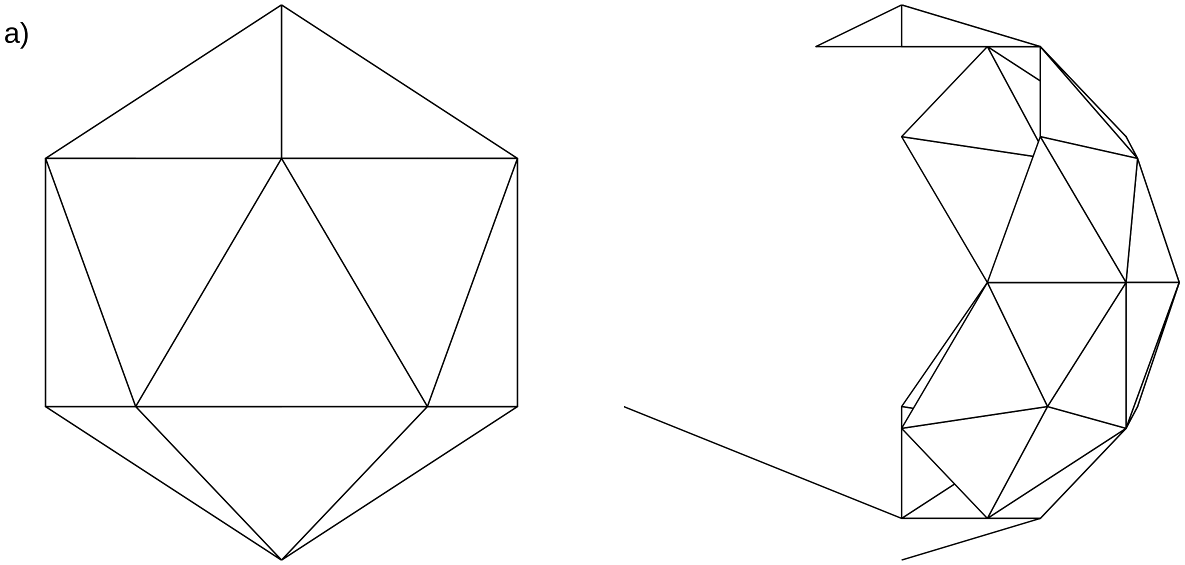
$$f(\Delta\theta) = \cos\left(\frac{\Delta\theta}{2}\right)\sec\left(\frac{3\Delta\theta}{2}\right) \qquad \text{where } \Delta x_n = \Delta x_0 \frac{\cos\left(\frac{(2n-1)\Delta\theta}{2}\right)}{\cos\left(\frac{(2n+1)\Delta\theta}{2}\right)} \tag{4.5 i}$$

$$f(\Delta\theta) = \cos(\Delta\theta) \qquad \text{where } \Delta x_n = \Delta x_0 \cos(n\Delta\theta)\ . \tag{4.5 ii}$$

Some percentage errors in the coefficient of the known ϕ-values are given for both methods of projection in tables 4.1 and 4.2. The correct value for each coefficient is taken as the coefficient from the uniform grid case.

These tables show that both methods of projecting the circle on to the line are second order accurate as the resolution is increased. Method i. is about twice as accurate as method ii. when using a second order polynomial but the errors are similar when using a fourth order polynomial. In both cases the accuracy is lower when using the higher or [illegible]

$\Delta\theta$	Percentage error in coefficients					
	Method i.			Method ii.		
	ϕ_{-1}	ϕ_0	ϕ_1	ϕ_{-1}	ϕ_0	ϕ_1
10°	1.5427	0.0000	1.5427	0		


a)

bisection and projection can be continued to form a hierarchy of triangular grids on the sphere, the dual grid of each of these triangular grids forms another in the hierarchy of icosahedral-hexagonal grids used in this work. The new vertices added to the grid are each surrounded by six triangles and so are associated with hexagons on the dual grid. There are still twelve pentagons at the vertices from the original icosahedron. The dual grid formed after five successive bisections and projections is shown in Figure 4.2 d, five of the twelve pentagons can be seen still showing the underlying icosahedron.

This grid is not symmetric about the equator and because of this numerical results will not be symmetric when symmetric conditions are used. Heikes and Randall [10, 11] suggest rotating the southern hemisphere through 36° after the first subdivision of the triangles to create a symmetric grid. This 'twisted where r_i is the distance between the midpoint of an edge and the midpoint of the arc joining the cent

N_{GRID}	Untweaked grid		Tweaked grid	
	${\frac{A_I}{A_J}}_{AVE}$	${\frac{A_I}{A_J}}_{MIN}$	${\frac{A_I}{A_J}}_{AVE}$	${\frac{A_I}{A_J}}_{MIN}$
1	1.000	1.000	1.000	1.000
2	0.941	0.884	0.942	0.884
3	0.949	0.902	0.978	0.931
4	0.960	0.895	0.990	0.944
5	0.973	0.895	0.995	0.944
6	0.983	0.895	0.997	0.945
7	0.991	0		

finding a stencil for and (for all but the zeroth order scheme) the five or six surrounding cells. On the triangular grid we continued adding cells that shared one or more edges with cells already in the stencil. In this case we are going to restrict the number of cells added at one time by only adding cells that share two or more edges with those in $H_k \mathbf{x} \; {}_k \qquad {}_k \quad H$

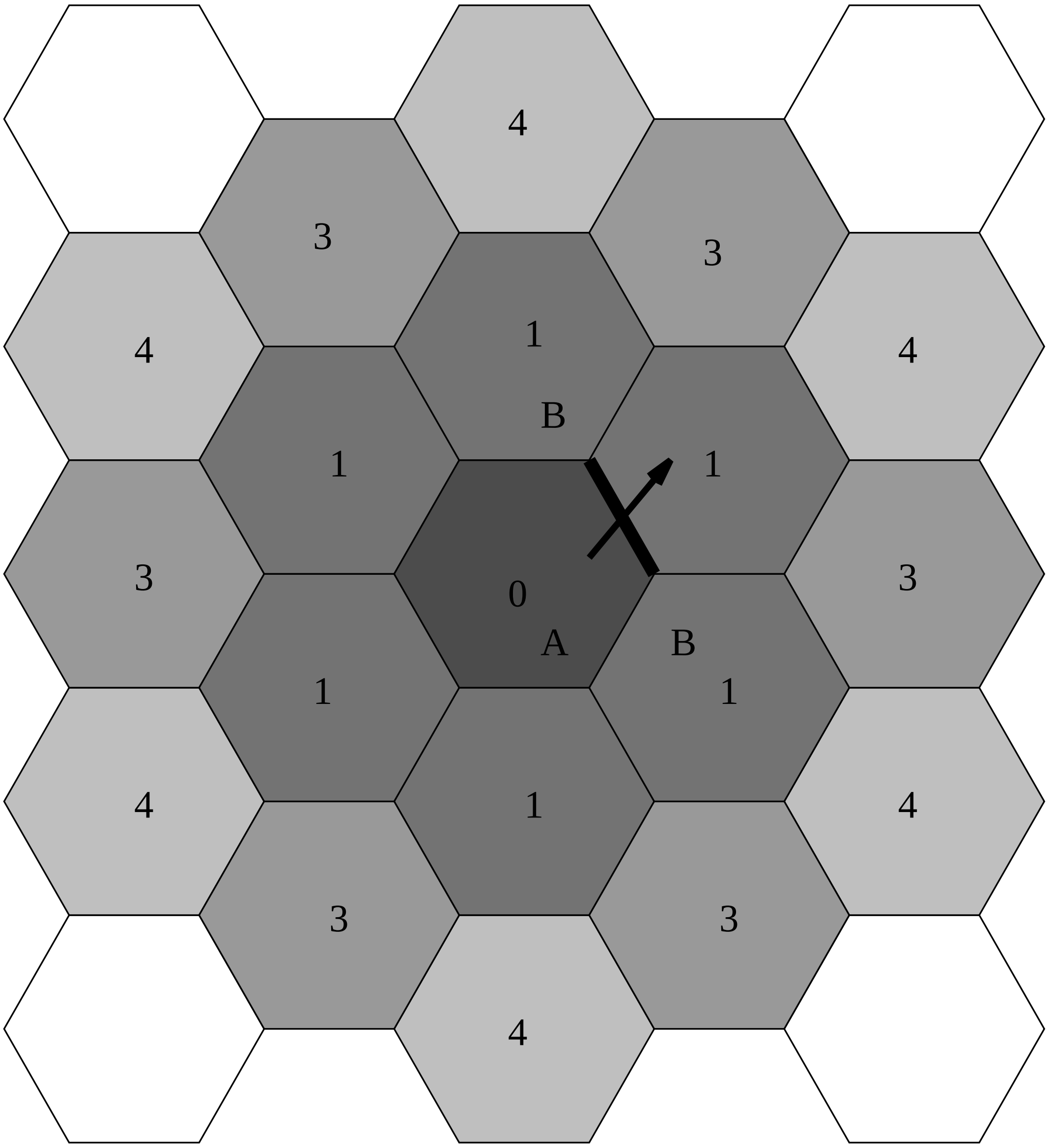


Figure 4.3: Examples of the stencils chosen over which to fit the desired polynomial. The numbers shown are the order of the polynomial for which the cells will need to be added to the stencil.

check that we have not altered their b

difference between cell centres is used to find the tangential component, the values at the vertices are found by interpolation and then differenced to find the normal component of the wind. For some test cases there will also be a divergent component of the wind which can be found from a velocity potential, χ, in a similar way.

4.5.1 Flow angle dependence

Many numerical schemes that use dimensional splitting for two dimensional flows also have a dependence on the angle of the flow compared to the grid. With an evenly spaced grid and a two-dimensional advection scheme there should be little dependence of the numerical solution on the angle of flow. On a regular latitude-longitude grid advection around latitude circles is performed at a constant resolution around each latitude. Advection over the poles is performed over a grid of variable resolution around each circle of constant wind speed. These differences will lead to different numerical results, their significance depending on the numerical method being used. On the icosahedral-hexagonal grid the grid spacings are uniform in all directions so we would not expect this difference to occur.

In order to test if there is any flow angle dependence in the numerical solutions of this scheme we shall run tests with the same initial conditions and a constant rotational flow at different angles to the grid. The initial condition is defined as being one in the region $-15° < \lambda,\ \varphi < 15°$ and zero elsewhere. The winds are given as

$$u = u_0 \left(\cos\varphi\cos\alpha + \sin\varphi\cos\lambda\sin\alpha\right) \ , \tag{4.11$_u$}$$

$$v = -\, u_0 \sin\lambda\sin\alpha \ , \tag{4.11$_v$}$$

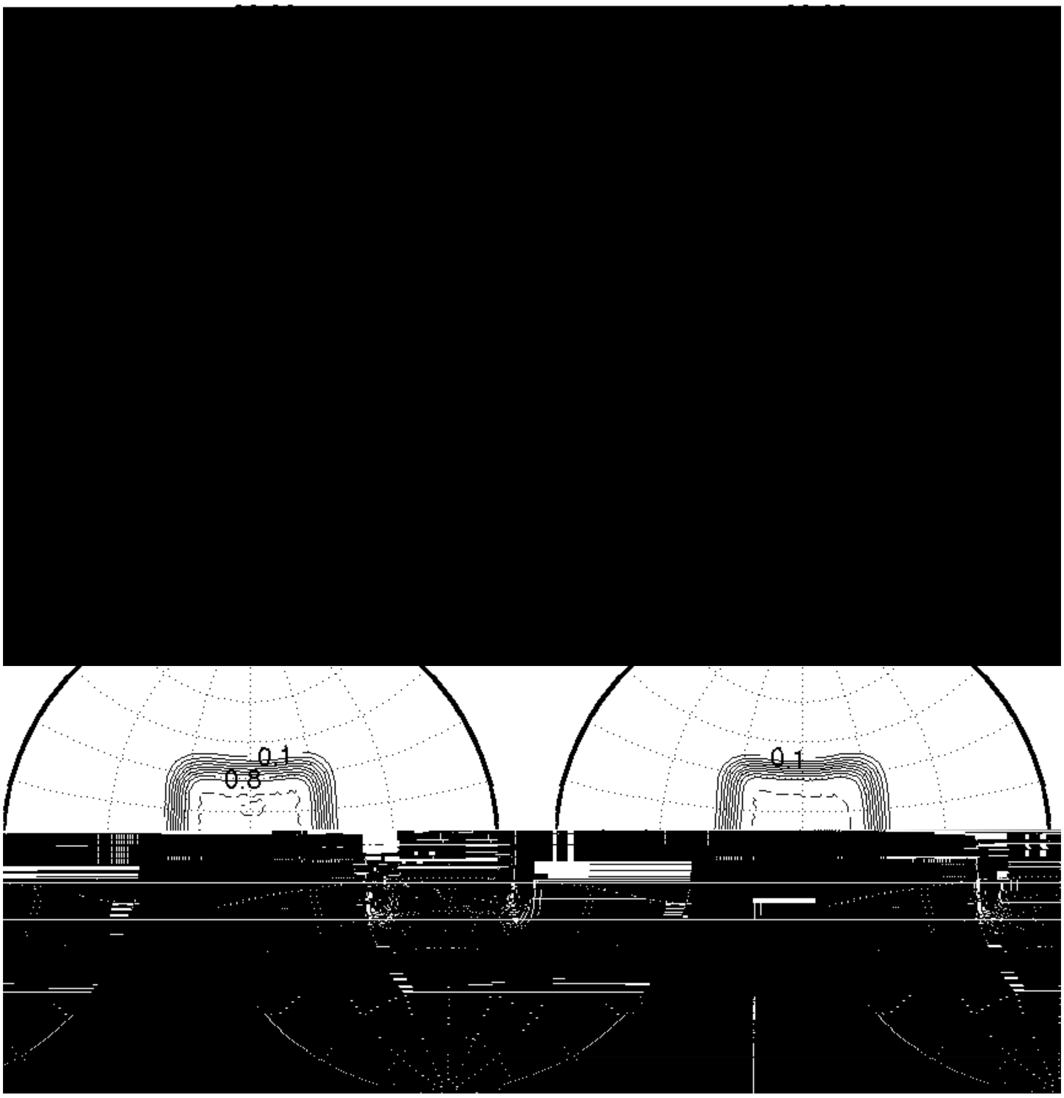
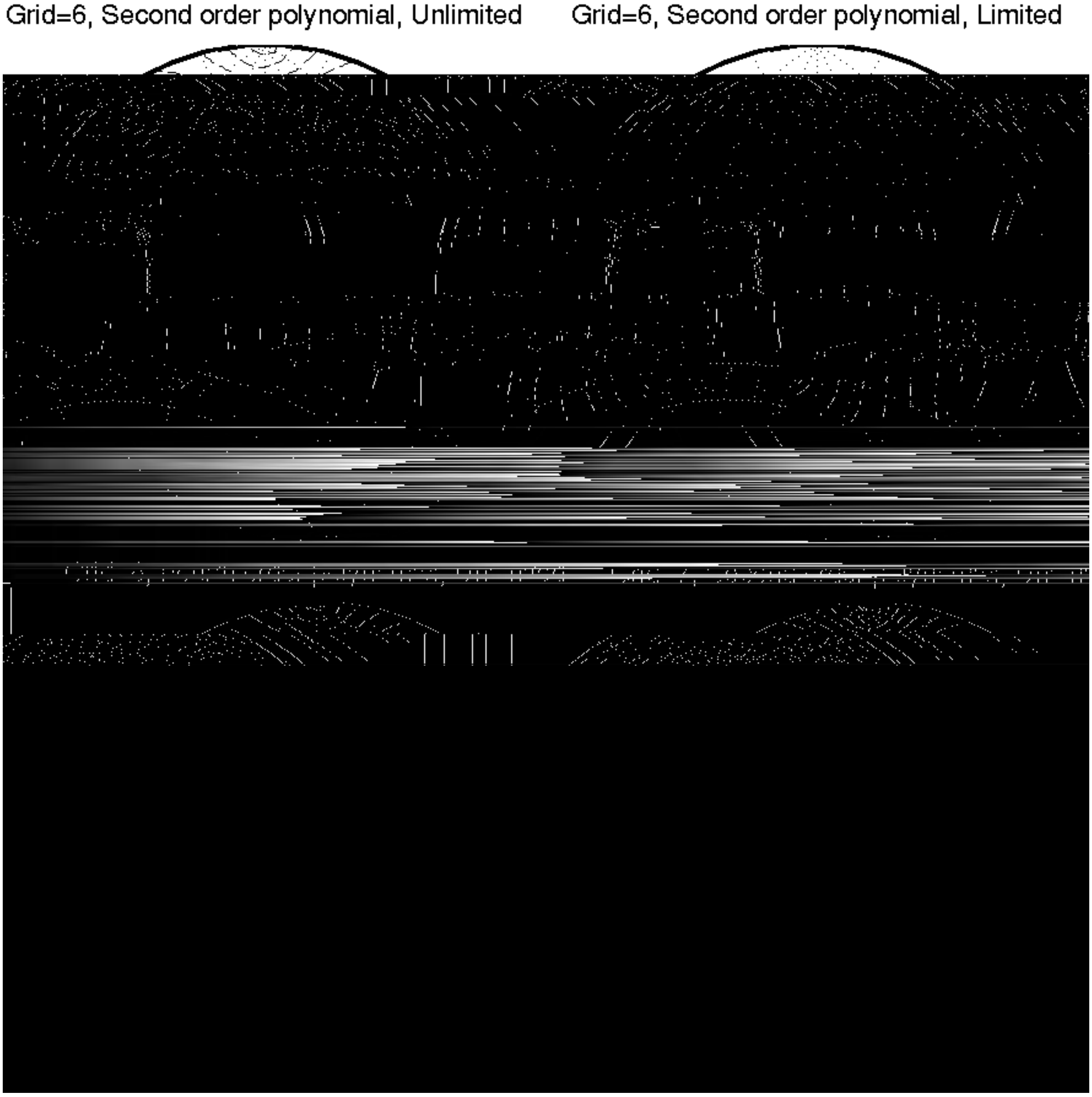


Figure 4.4: Numerical solutions of the advection of a step region around a sphere at an angle α to the equator. The numerical scheme uses a second order polynomial and the flux limiter on grid number 6. The contour interval is 0.1 and contours at 1 are dashed.

profile across the poles (α =90°). This test was run on grids 5, 6 and 7 using schemes based on zeroth to sixth order polynomials both with and without the flux limiter. The same error measures as in previous chapters have been used to measure the error with the CPU runtime is taken as the measure of the computational cost.

Figure 4.5 shows a selection of the solutions generated by these schemes. Using a second order scheme without the limiter on grid 6 gives a solution



Grid=6, Second order polynomial, Unlimited

Grid=6, Second order polynomial, Limited

Figure 4.5: Numerical solutions of the advection of a step region ar

and generally improves the accuracy of the scheme. The improvement in the error is large compared to the increased computational cost for schemes up to and including third order. Increasing the order of the scheme beyond this does improve the accuracy but also increases the cost disproportionately. These higher order schemes may still be more cost effective than increasing the resolution of the grid since this can only be done in large increments.

4.5.3 Test Case 1: Advection of a cosine bell over the pole

This is the first of the seven Williamson test cases for numerical approximations to the shallow water equations in spherical geometry. It is included here since it is a test that involves only pure advection. The shallow water model that is used is described in the next chapter. The model is run in this case with the same wind field prescribed at each time step so that only the advection part of the model is used. The method for generating advection schemes that has been developed in this work has been incorporated into this model.

The test is the advection of a smooth cosine bell of radius $R = a/3$ once around the sphere at various orientations of the advecting winds. The cosine bell is defined as

$$h(\lambda, \varphi) = \begin{cases} \frac{h_0}{2}\left(1 + \cos\left(\frac{\pi r}{R}\right)\right) & r < R \\ 0 & r \geq R \end{cases} \tag{4.13}$$

where $h_0 = 1000\text{m}$ and r is the great circle distance of (λ, φ) from $(\lambda_c, \varphi_c) = (3\pi/2, 0)$:

$$r = a\cos^{-1}\left(\sin\varphi_c \sin\lambda + \cos\varphi_c \cos\varphi \cos(\lambda - \lambda_c)\right) \ . \tag{4.14}$$

The wind field is the same as that used for the two previous test, defined

Figure 4.7: Numerical (solid) and exact (dashed) solutions for test case 1 with α =90°, advection across the poles.

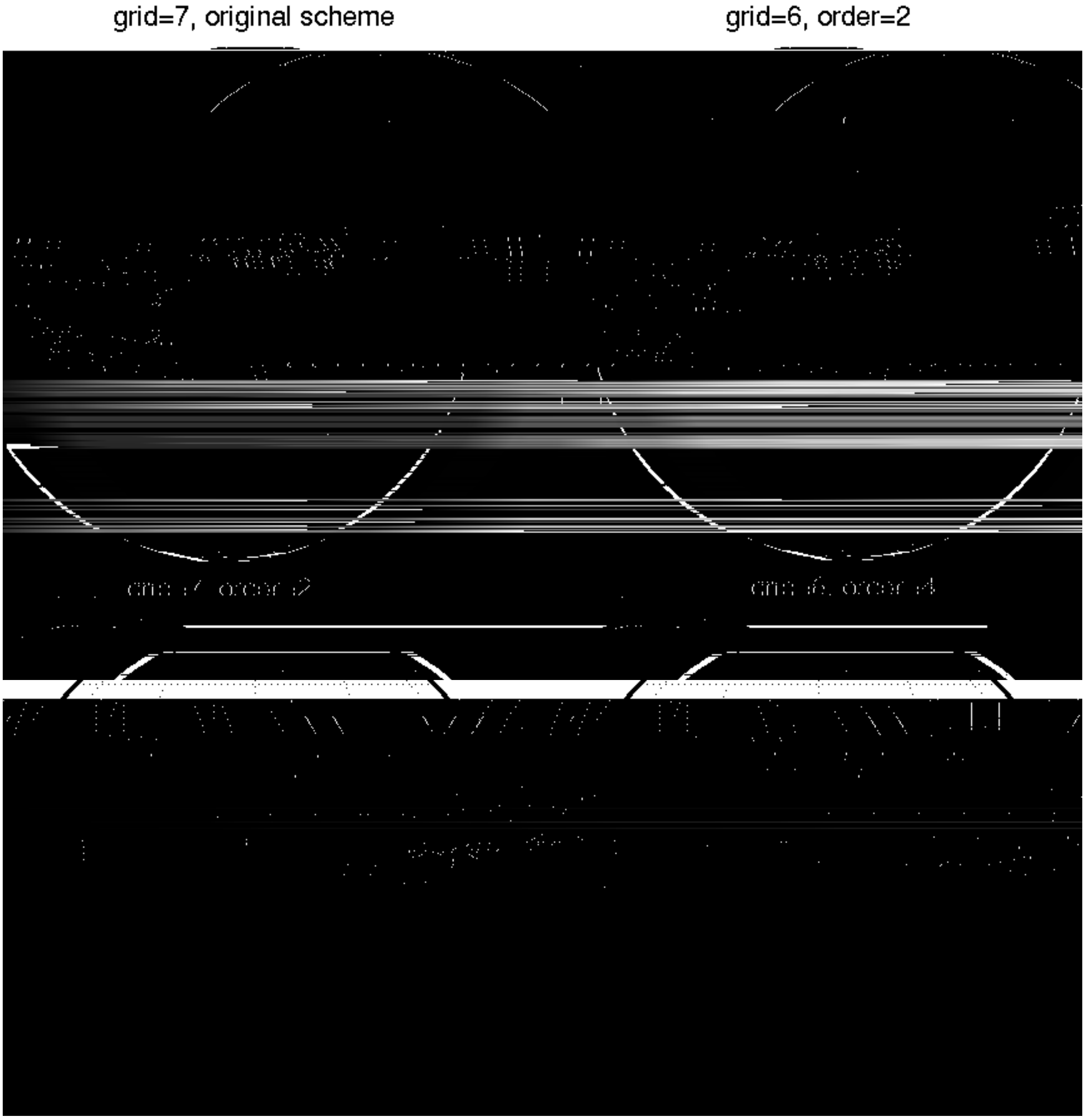


Figure 4.8: Difference between numerical and exact solution for test case 1 with α =90°, advection across the poles.

approximations. Increasing the order of the polynomial to four reduces the rate at which the error grows by over one half in this case. These numerical errors show that the second order scheme on grid 7 does give slightly more accurate results than the fourth order scheme on grid 6 though at a much increased computational cost.

All these errors grow smoothly with time showing that the scheme does not encounter problems with any particular part of the grid, i.e. across the poles. The slight oscillations in the L_∞ error occur as the peak of the cosine bell crosses cell boundaries and cell centres. Although the results presented here only show one orientation of the flow, the results are the same for all the other orientations tested, including those of the Williamson test cases.

4.6 Summary

The method developed over the previous chapters has been successfully applied to an icosahedral-hexagonal grid on the sphere. The method itself did not require any adjustments but we did make a local plane approximation to the spherical grid for this to be the case. This idea was investigated in one dimension and shown to make only a small difference to the schemes generated when grid cells were only a few degrees of arc.

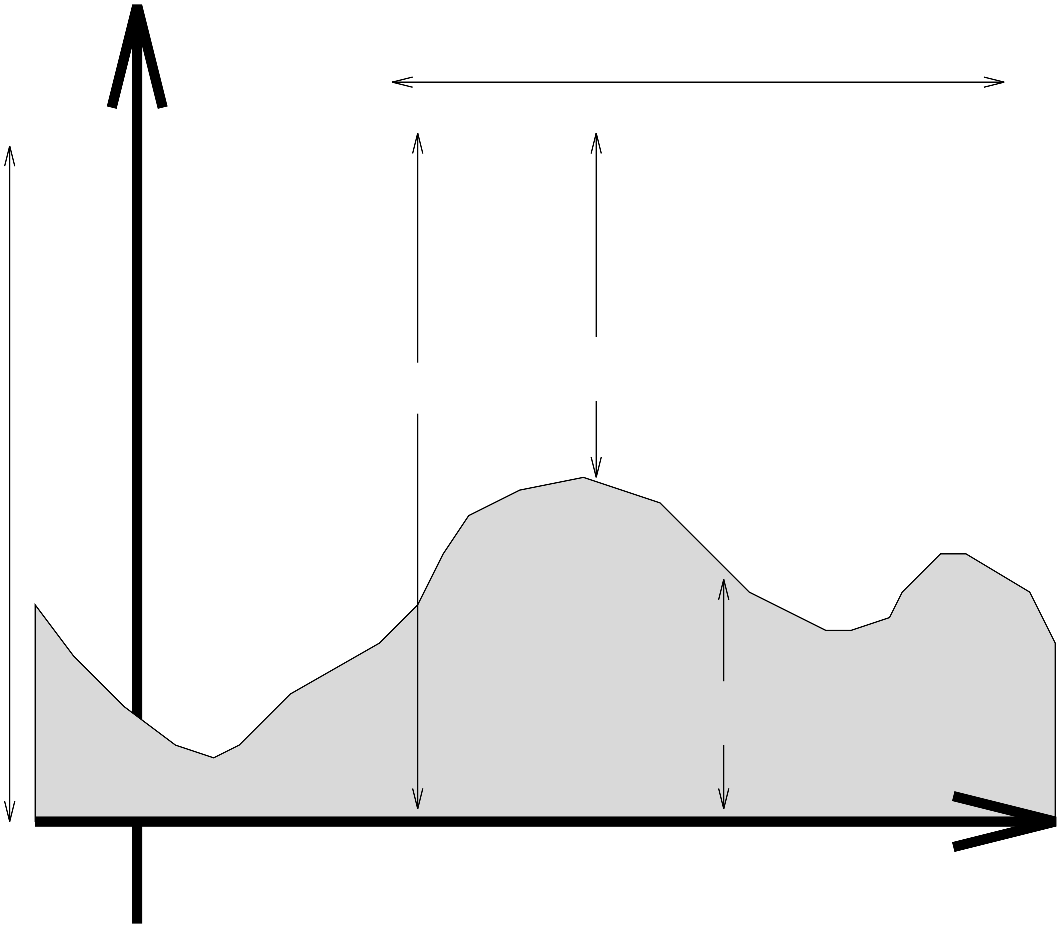
The results gained from the application of the method to the icosahedral-hexagonal grid were similar to those of previous tests on square and triangular grids. The changes in accuracy of the results as the order of the scheme is revs

Chapter 5

Shallow Water Model

5.1 Introduction

In the last chapter we applied the method of generating advection schemes developed in earlier chapters to an icosahedral-hexagonal grid on the sphere. The advection schemes generated by the method were s



defined a

> "One possible reason for this may be the lack of adequate advection schemes."

PV has also recently been used by Wlasak [60] as a diagnostic tool to describe balance in data assimilation.

The model used in this work is the same as that used by Thuburn [52, 53]. The original advection scheme used in that model was based on a quadratic interpolation over regular hexagons that is third order accurate on the plane. This method is, like the new schemes we shall use, only formally first order accurate on the grid used. Tests have been run using advection schemes based on second and fourth order polynomial interpolation using the new method. As before, the new schemes are referred to according to the order of the polynomial used for interpolation, not the formal order of accuracy.

The remainder of the model is identical to that of Thuburn described in detail in [52], only a brief description of the rest of the model is given here. The Laplacian of a field, ϕ, is given by,

$$\frac{1}{A_k}\sum_i \frac{e_i}{d_i}(\phi_i - \phi_k) \tag{5.6}$$

where i runs over the edges and neighbouring cells of cell k, e_i and d_i are the length of edge i and distance between the centers of cells k and$i$. This is a second order accurate approximation on regular hexagons but is less accurate on the distorted hexagons of this grid. Poisson's equation is solved using a multigrid elliptic solver that uses the hierachy of grids described in the last chapter. A semi-implicit time step is used which introduces a number of subtleties that are dealt with in [52].

The other difference from the results reported in Thuburn [52, 53] is that the grid was 'tweaked' (but not 'twisted') as described in the last

The other parameter values used are $u_0 = 2\pi a/12$ and $gh_0 = 2.94 \times 10^4$ $\mathrm{m}^2\mathrm{s}^{-2}$.

Since this is a steady state of the shallow water equations we would

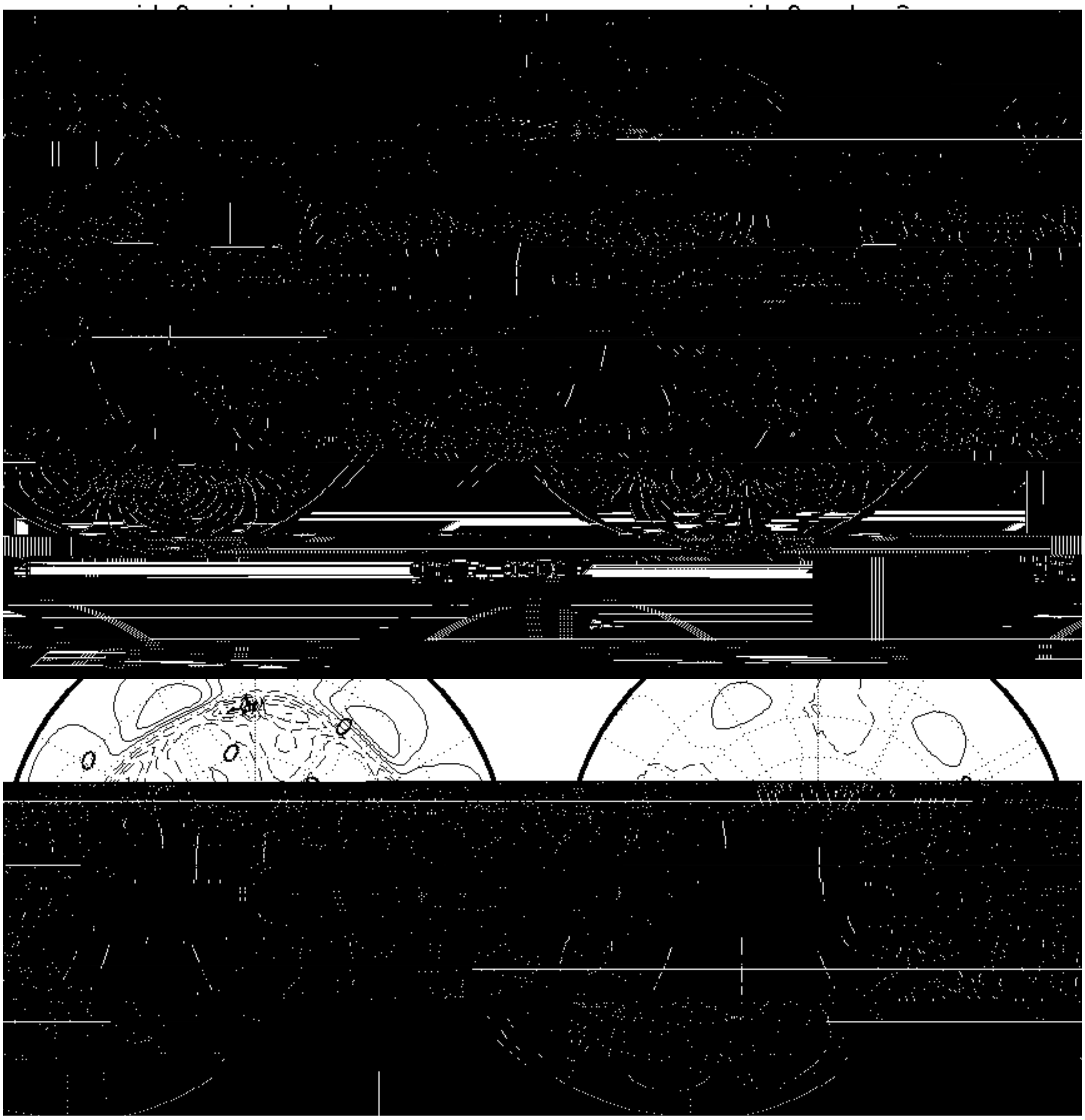


Figure 5.2: Difference between true and numerical height fields for the northern hemisphere after a five day run of test case 2 ($\alpha = 0$). Contour interval is 0.25 m.

maximum values of the error still correspond with the pentagonal cells in this case but have a slightly smaller magnitude. The same comments can be made when comparing the errors of the second and fourth order schemes. The pattern of error is different again but the five fold symmetry is still present and the peak errors occur over the pentagons. The second order scheme run on grid 7 shows a similar pattern of errors to that on grid 6 but greatly reduced in magnitude.

The five-fold symmetry of the errors is not unexpected because of the five-fold symmetry of the grid. The greatest irregularity in the size of the grid cells occurs at the pentagons so we would expect both the greatest difference between the continuous and discrete steady states and the poorest approximation to

$$\frac{\partial ()}{\partial t} = 0 \ , \tag{5.10}$$

to occur here. This would lead to the larger errors near the pentagonal cells, particularly in a flow where each lies in a region upwind of another. The advection scheme being used will affect the numerical steady state solution. The difference between the three different cases on grid 6 show that the interactions between the grid and the advection scheme can cause significant differences in the results.

The time evolution of the normalised global error norms for the height field is shown in Figure 5.3. The errors from the original scheme are slightly larger than those from the un-tweaked grid shown in Thuburn [52], figure 18 . There are fewer oscillations in the error as the model settles from the continuous balanced state to the model one on the tweaked grid. These oscillations are more in evidence when the new second order scheme is used but overall the error grows more slowly using the new

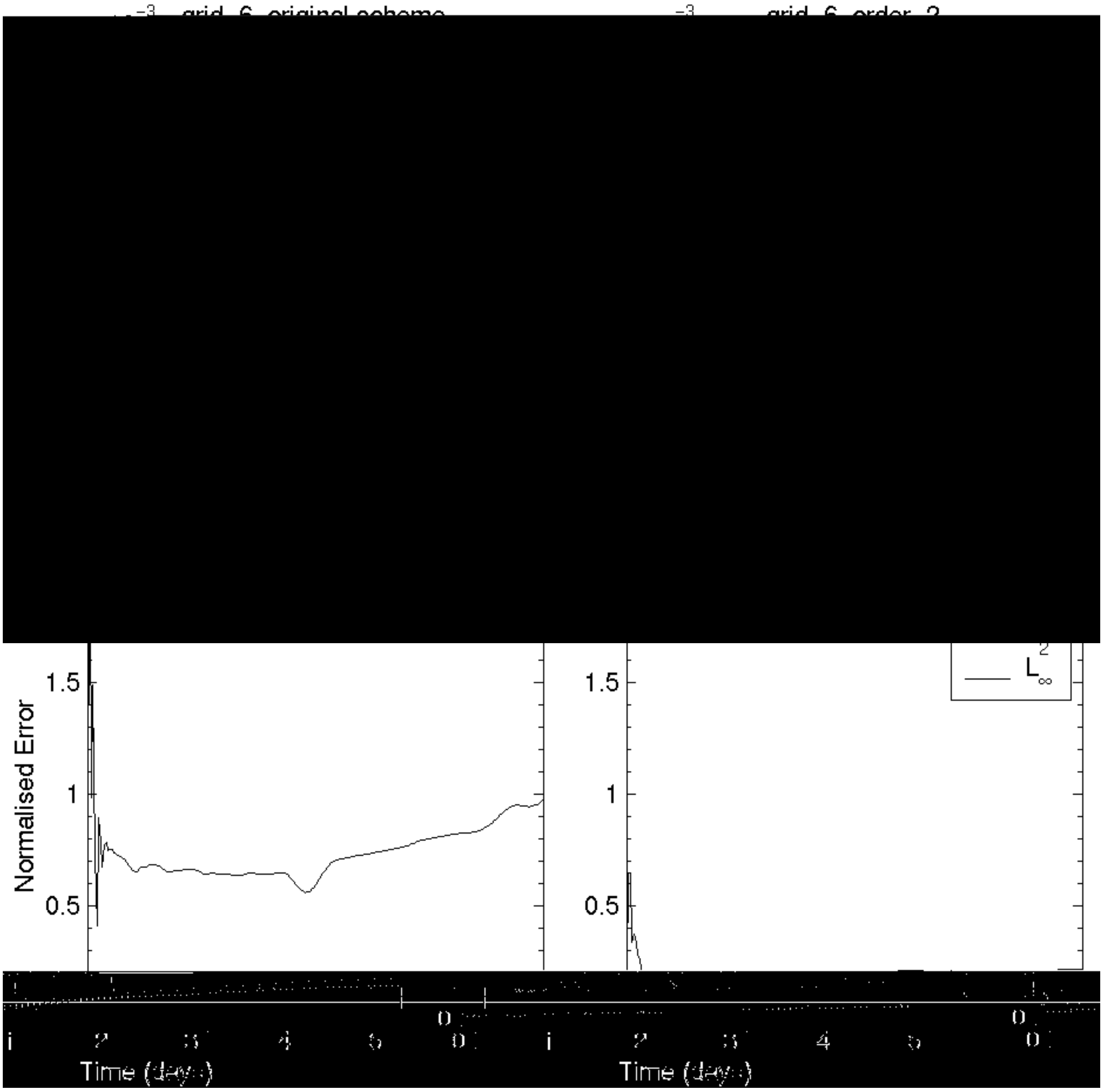


Figure 5.3: Normalised global errors in the height field for a five day run of test case 2 ($\alpha = 0$). Errors sampled at each time step (30 mins.).

scheme. These L_1 and L_2 errors are roughly halved when the order of the scheme is increased to four, and quartered if the grid resolution is increased to grid 7 instead. The L_∞ error shows a similar decrease in

The global height and wind errors are very similar to these results for different angles between the rotational and grid poles. The five-fold symmetry is distorted when $\alpha = 0.05$ and destroyed when α is close to $\pi/2$, as in the original model. The magnitude of the errors when using the fourth order scheme is comparable with the model of Heikes and Randall [12] at this resolution. Where the errors in this case were growing, those of Heikes and Randall oscillate in time about a uniform value. The spectral model of Jakob et al. [1 rotating at an angle α to the earths rotation the coordinate systems are related by

$$\sin\varphi' = \sin\varphi\cos\alpha - \cos\varphi\cos\lambda\sin\alpha \ , \tag{5.12}$$

$$\sin\lambda'\cos\varphi' = \sin\lambda\cos\varphi \ . \tag{5.13}$$

The correct quadrant for λ' can be found by ensuring

$$\sin\varphi = \sin\varphi'\cos\alpha + \cos\varphi'\sin\alpha\cos\lambda' \ . \tag{5.14}$$

The Coriolis parameter for the two systems is

$$f = 2\Omega\left(-\cos\lambda\cos\varphi\sin\alpha + \sin\varphi\cos\alpha\right) \ , \tag{5.15}$$

$$f' = 2\Omega\sin\varphi' \ . \tag{5.15'}$$

The analytic height field in the prime coordinate that gives a steady state solution must satisfy

$$\frac{(u')^2\tan\varphi'}{a} + \frac{g}{a}\frac{\partial h'}{\partial\varphi'} + fu' = 0 \ . \tag{5.16}$$

The analytic height field is difficult to find exactly from this equation, instead numerical integration is used to find an accurate approximation to h,

$$h = h_0 - \frac{a}{g}\int_{-\frac{\pi}{2}}^{\varphi'}\left(2\Omega\sin\tau + \frac{u'(\tau)\tan\tau}{a}\right)u'(\tau)\mathrm{d}\tau \ . \tag{5.17}$$

The background height is again given by $gh_0 = 2.94 \times 10^4\ \mathrm{m^2s^{-2}}$.

5.4.2 Results

> "... presumably about a balanced state of the discrete equations which is not exactly the same as the initial state." (Thuburn [52])

Figure 5.5 shows the difference between the true and model solutions for the same four advection schemes as test case 2. The original scheme used with the tweaked grid shows no obvious difference from using the scheme on the un-tweaked grid. When the new second order scheme is used the difference between the true and numerical solutions is smaller but the five fold symmetry is much clearer to see in a broader region in the error field. If the fourth order scheme is used the height differences at day five are larger than those for the second order scheme. A possible reason for this can be seen in the time evolution of the global error measures which oscillate in time with slightly different phases. When the second order scheme is used on grid 7 the symmetry of the grid is less visible and the overall error is smaller

Figure 5.6 shows the global L_1 , L_2 and L_∞ errors in the height field for these four runs. All cases exhibit the oscillations in the errors seen in the original model. The change in the grid has made little change to the height errors when the original scheme is used. The oscillations in the error are more irregular on the tweaked grid but have a similar magnitude and period. The oscillations in the global error are very similar in form when the new second order scheme is used but the error is smaller in magnitude. Using the fourth order scheme reduces the amplitude of the oscillations slightly but does not improve the magnitude of the errors. Increasing the resolution of the grid does reduce the magnitude of the error but the oscillatory nature of the errors and even the pattern of the oscillations are very similar.

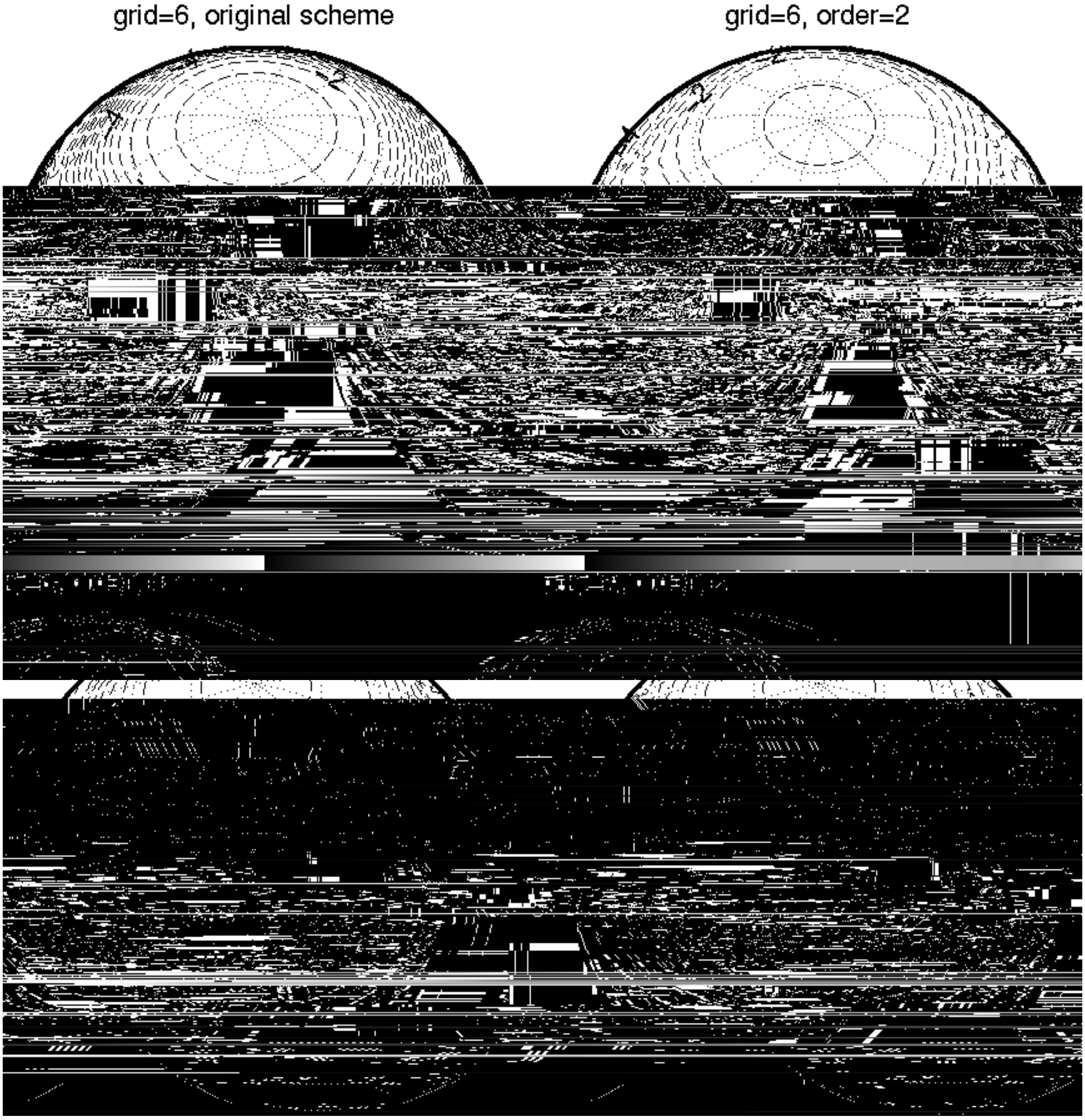


Figure 5.5: Difference between true and numerical height fields after a five day run of test case 3 ($\alpha = 0$). Plots are centred on $(\frac{3\pi}{2}, \frac{\pi}{2})$ Contour interval is 0.5 m.

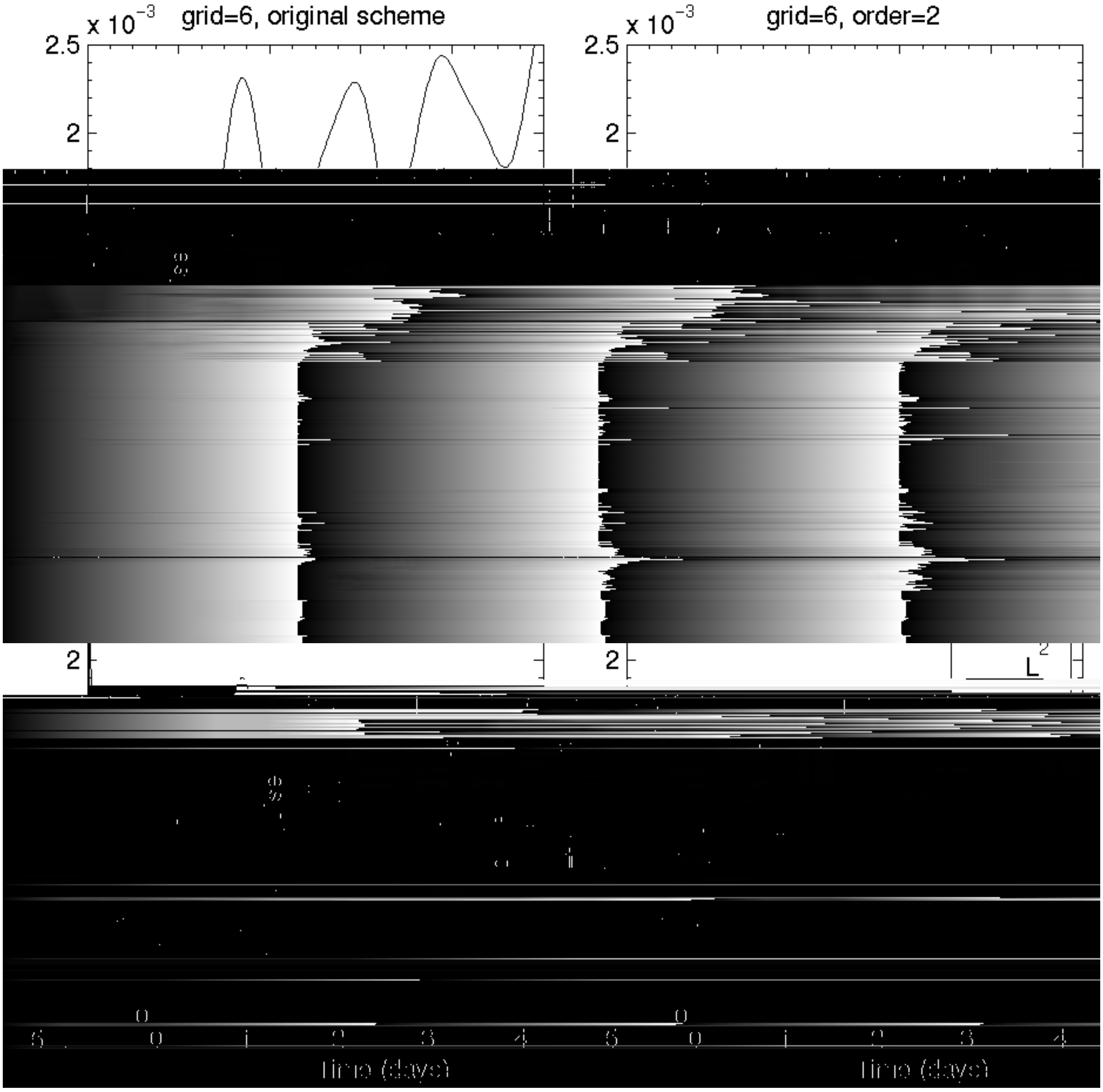

x 10^-3
grid=6, original scheme
2.5
2
x 10^-3
grid=6, order=2
2.5
2
2
2
L^2
0
0
5
0
1
2
3
4
5
0
1
2
3
4
time (days)
time (days)

Figure 5.6: N

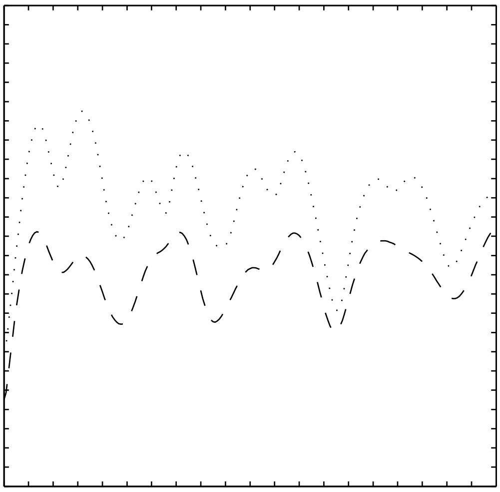


Figure 5.7 shows these same error measures for the wind field. These are also oscillatory in nature and again show the same patterns and magnitudes in the L_1 and L_2 errors for each different advection scheme. This strongly suggests that these errors are not due to errors caused by the advection scheme. The similarity of the results on grid 7 further suggests that these errors are not related directly to the grid resolution. Two possibilities for the cause of this error are the time truncation errors in the divergence equation and initial errors in the discrete height field. The initial discrete height field is unlikely to give a steady state solution, the model state may therefore be oscillating around some discrete steady state. The errors in modelling the small amount of divergent flow caused by the discrete height field may also be causing the errors. Comparing the advection schemes in this case using figures 5.6 and 5.7 cannot therefore go much beyond stating that they all do a good enough job not to seriously affect the results of this test case.

The results for these schemes are very similar when the angle α is changed to $\pi/3$. The main difference is the disappearance of the five fold

where F_h is the height forcing given along with F_u and F_v in Williamson et al. [59]. The forcings of height, PV and divergence are applied to the fields at each time step. Numerical approximations to the curl and divergence operators are used as would occur in a real application of the model.

The time varying flow is given by

$$\tilde{u} = \bar{u} - \frac{\bar{\psi}_\varphi}{a} \ , \tag{5.20}$$

$$\tilde{v} = \frac{\bar{\psi}_\lambda}{a \cos\varphi} \ , \tag{5.21}$$

$$g\tilde{h} = g\bar{h} + f\bar{\psi} \ , \tag{5.22}$$

where

$$\bar{u} = u_0 \sin^{14}(2\varphi) \ , \tag{5.23}$$

$$g\bar{h} = gh_0 -$$

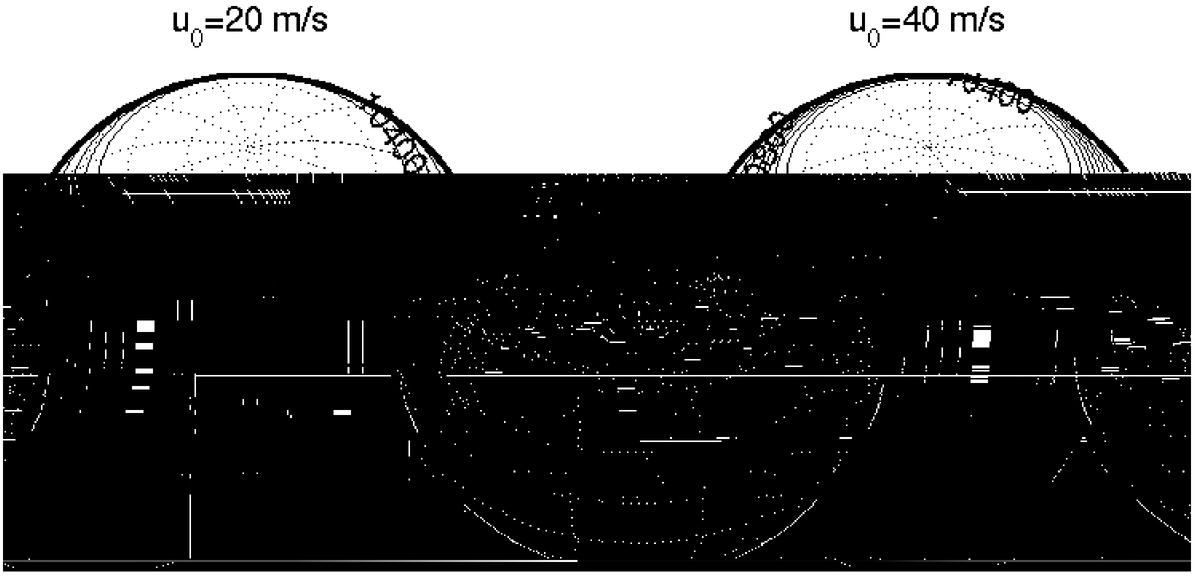


Figure 5.8: Exact solutions for test case of a forced non-linear system of a translating low after 5 days. Plots are centred on longitude of low centre. Contour interval is 100m

5.5.2 Results

Thuburn [52] suggested that the main error in the original model was the filling of the low centre

> "... consistent with a smoothing of the PV maximum (and to some extent the depth minimum) by the advection scheme."

An improved advection scheme should therefore cause less smoothing of the PV field and less filling of the low centre.

The exact solutions for both cases are shown in Figure 5.8 after 5 days. This plot and all others in this section have been rotated to the longitude of the low centre. Looking at the numerical solutions of the height fields shows little difference between the different schemes so instead we shall look at the differences between the true and numerical solutions shown in figures 5.9 and 5.10.

Figure 5.9 shows this difference for the case when $u_0 = 20$ ms^{-1} using the same four schemes used in previous tests. The results for the original

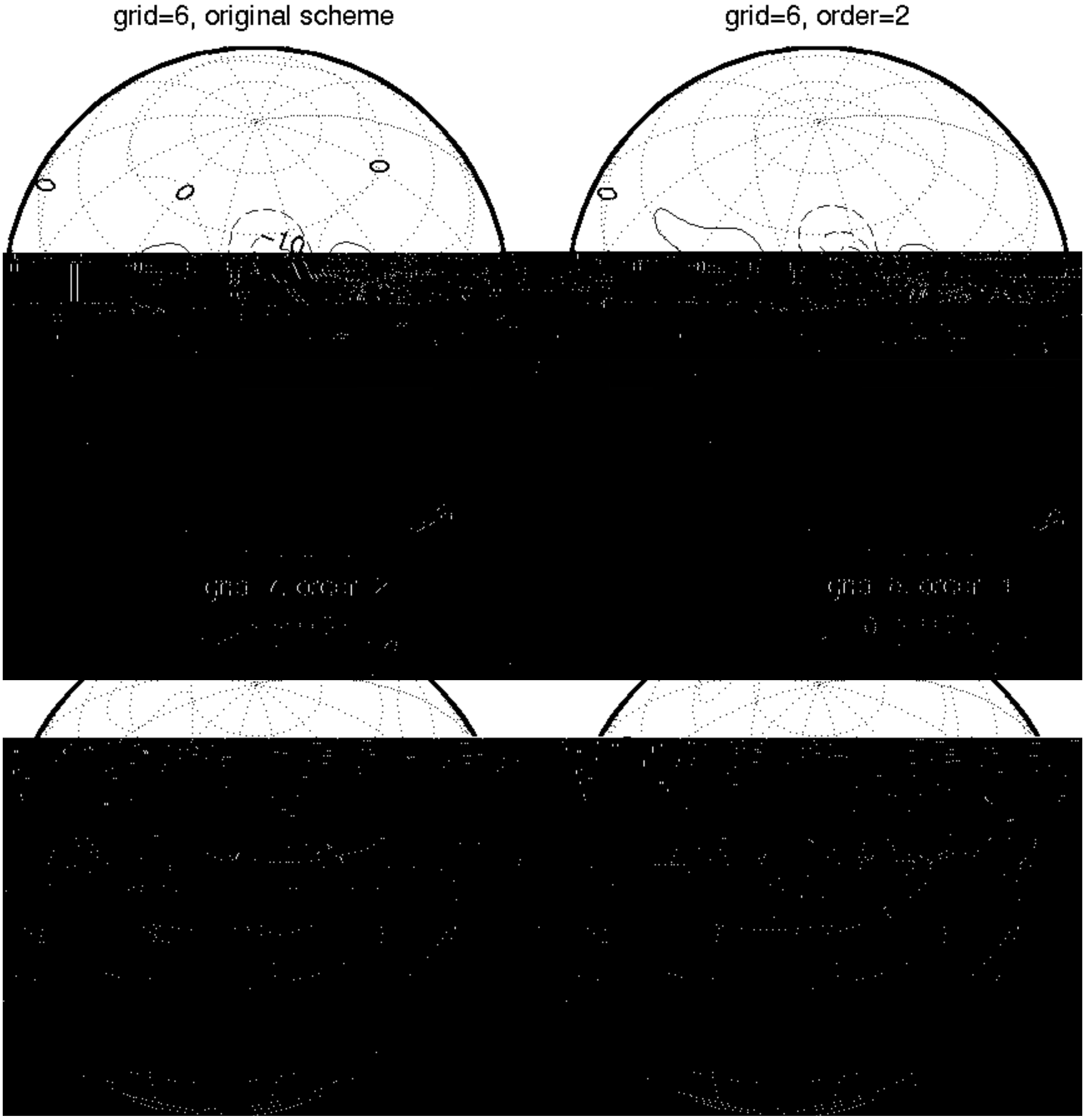
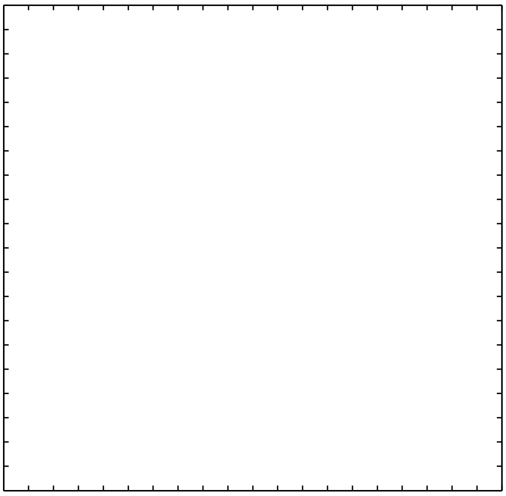


Figure 5.9: Difference between true and numerical height fields after a five day run of test case 4 ($u_0 = 20$ ms^{-1}). Contour interval is 5 m.

This suggests that using a fourth order polynomial for interpolation in the advection scheme captures all the features in the advected fields for this case. A second order polynomial captures many of these features but not all, as shown by the slow growth in errors for these cases. On the higher resolution grid using a second order polynomial the height errors do not grow significantly, suggesting that this combination of grid and scheme captures all the features of the advected profile. The results for the slower flow case are broadly the same, the only exception being a very slow growth of the error for the grid 7 case.

Figure 5.12 shows the global L_1 , L_2 and L_∞ wind errors for the same runs as before. The L_2 and L_∞ errors for the original scheme are very similar to those on the un-tweaked grid but the L_1 error is greatly reduced. This may again be because of a better representation of the source terms on the tweaked grid. Using the new second order scheme makes little difference to the results but the fourth order scheme does reduce the rate at which the errors grow. A greater reduction in this rate can be gained by increasing the resolution instead of the order of the scheme.

The results of the model of Heikes and Randall were poor and were not presented in full in [12]. From the comparisons that can be made, this model significantly outperformed that of Heikes and Randall. It correctly located, and produced less filling of the low centre. Comparison with the T42 spectral model show that the icosahedral model has some way to go before it can compete with the spectral model for this case. All the error measures of the spectral model are around a factor of ten smaller for this test. The spatial distribution of the errors for the icosahedral model does not oscillate in the same way as those for the spectral model but this is the only, small, advantage it has.



5.6 Test Case 5:

5.6.1 Zonal flow over an isolated mountain

This test case is

and our numerical one (see figures 5.14 and 5.15). This is an artifact of the spectral methods used in the spectral model which is not present in the model we are using and it will contaminate the measures of the error to an unknown extent.

5.6.2 Results

The original model performed well in this test, resolving the features of the large scale flow forced by the mountain at a lower resolution than the one used here. One problem that was noted was the significant drift away from zero of the global integral of vorticity. The suggested reason for this was the inconsistency between the updated divergence, δ^{m+1} and the divergent part of the wind $\mathbf{v}_{div}^{m+1}$ caused by the approximate solution of the elliptic equation

$$\nabla \cdot \mathbf{v}_{div}^{m+1} = \delta^{m+1} \ . \tag{5.29}$$

Changing the advection scheme is unlikis x [illegible] ro le g [illegible] t he t el [illegible] ng

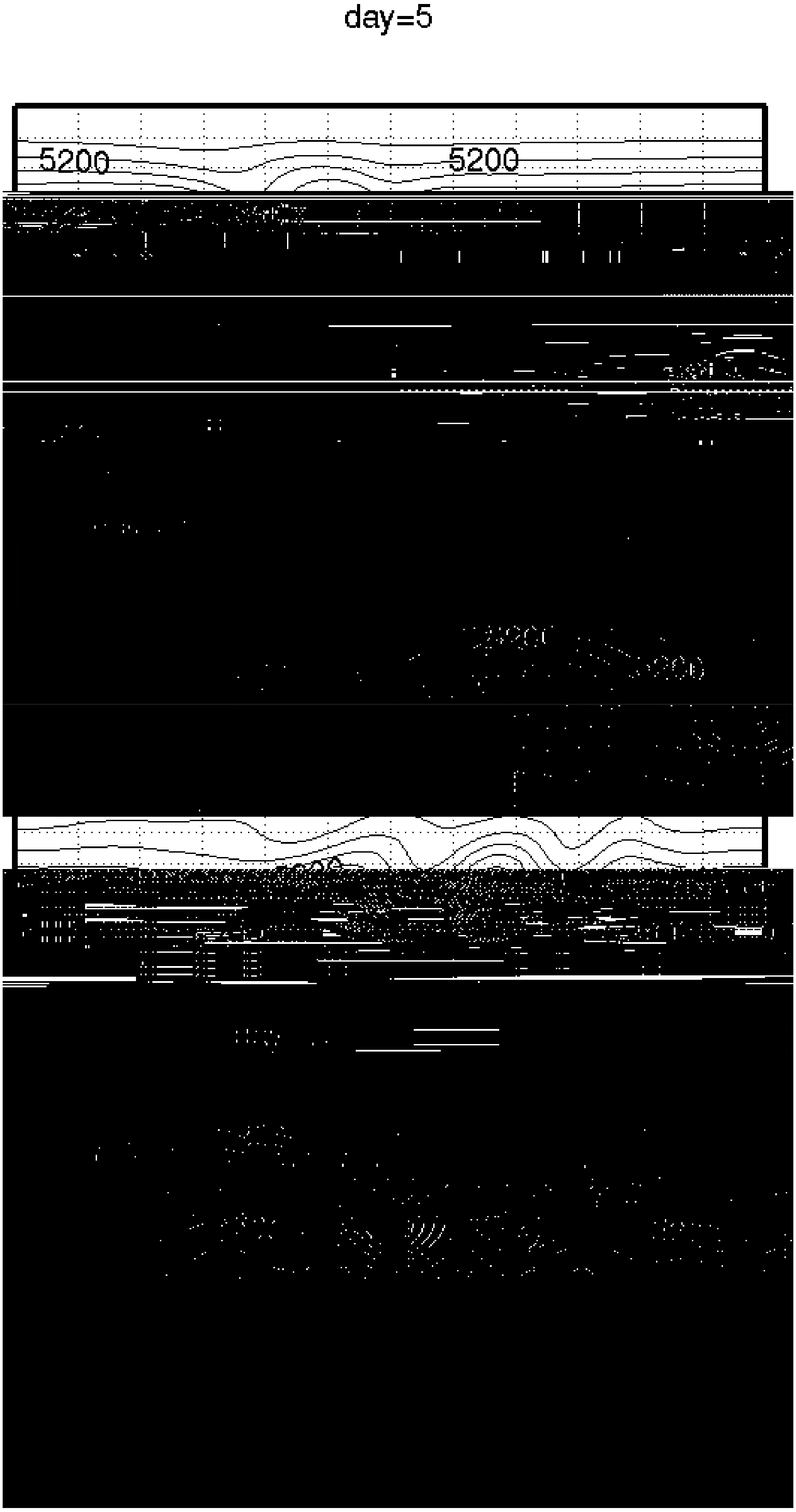


Figure 5.13: High resolution spectral solution of the height field for test case of flow over a mountain after 5, 10 and 15 days

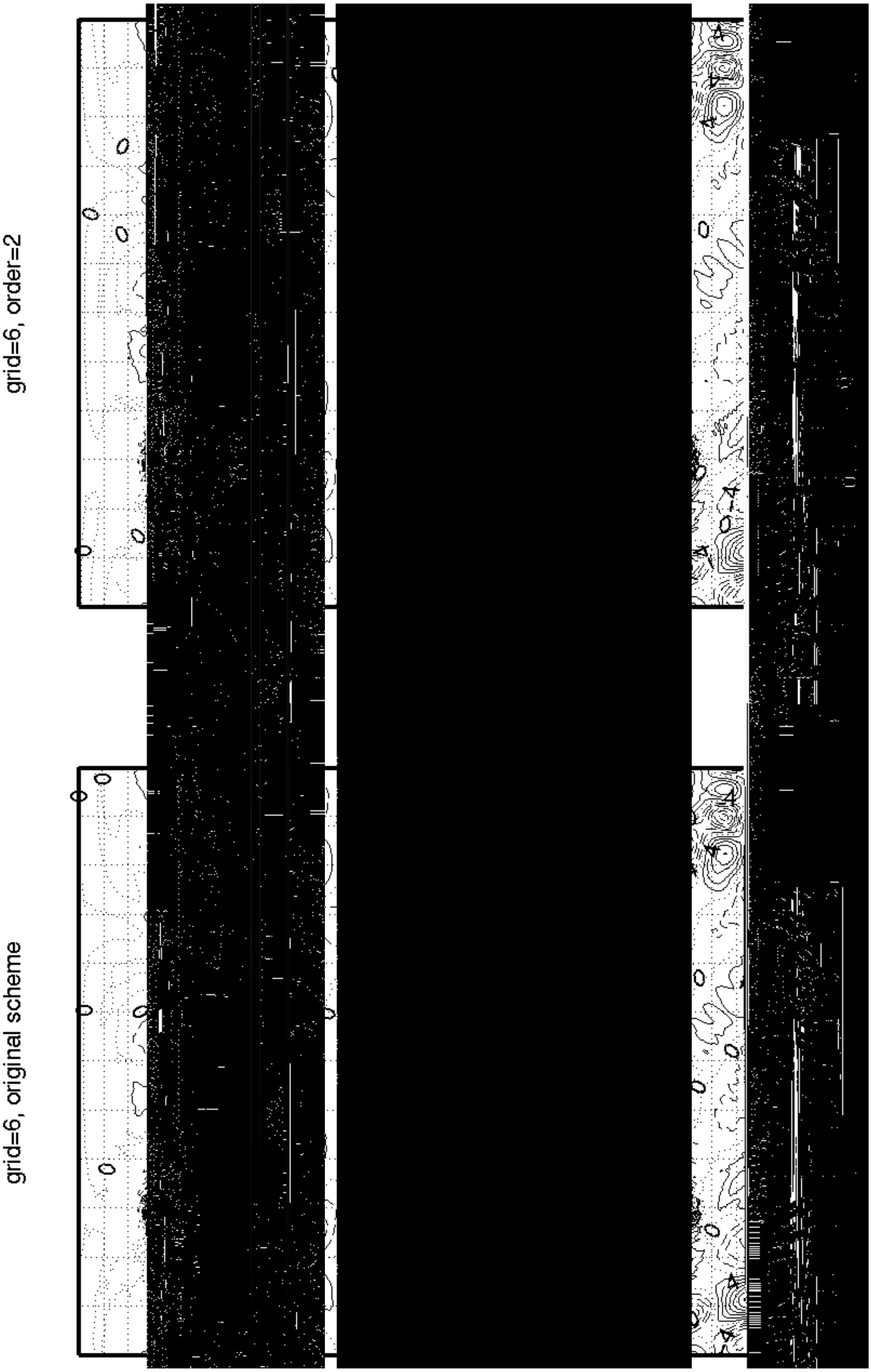


Figure 5.14: Difference between spectral and icosahedral height fields after a five days of a run of test case 5. Contour interval is 2 m.

such problems representing the mountain. After five days, Figure 5.14, the errors downwind of the mountain are smaller than those in other regions of the flow. This suggests that the advection in the flows generated by the mountain is well modelled but that other dynamic processes are not so well captured. In both regions, the differences between the different schemes is remarkes is re

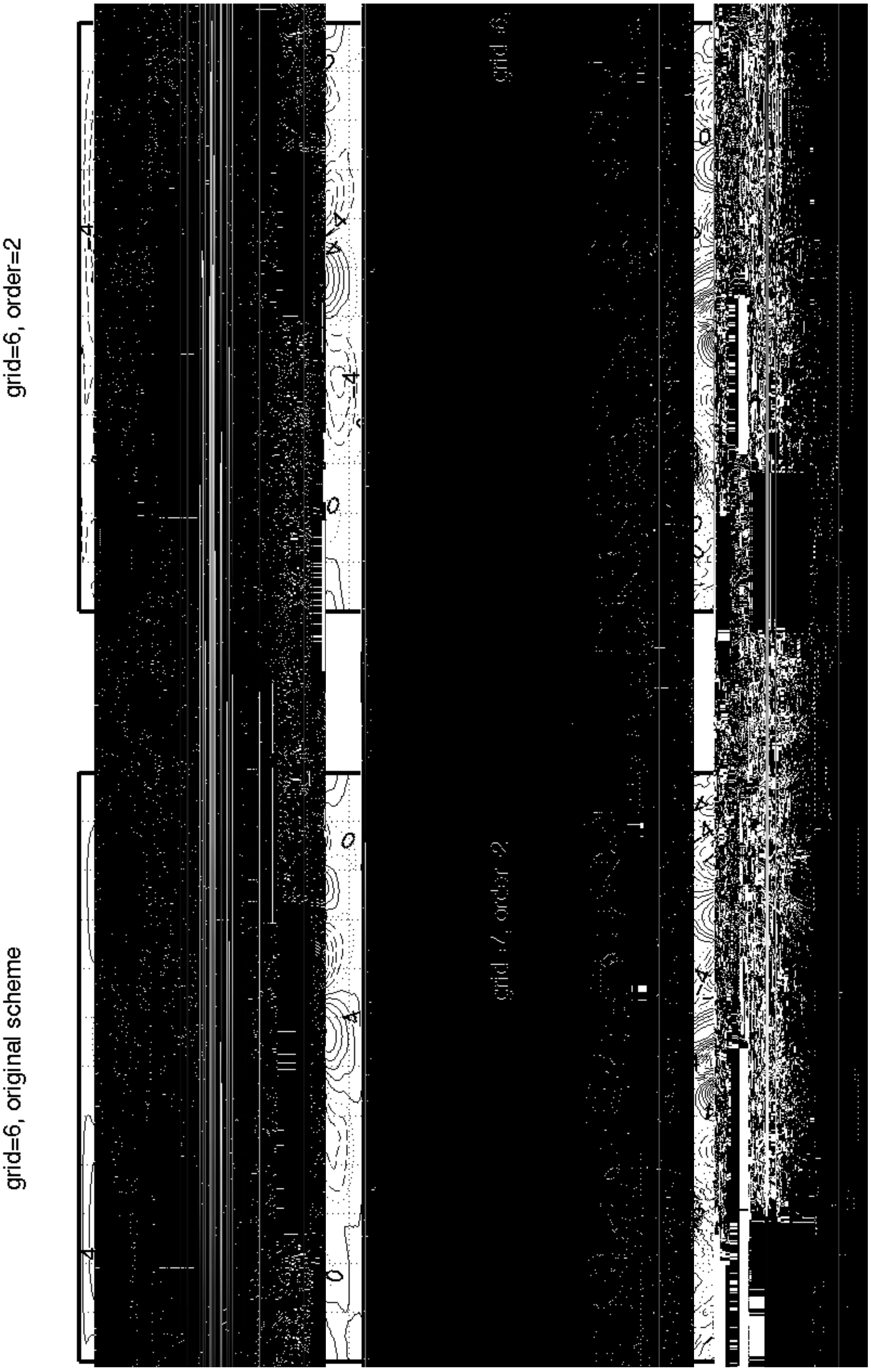
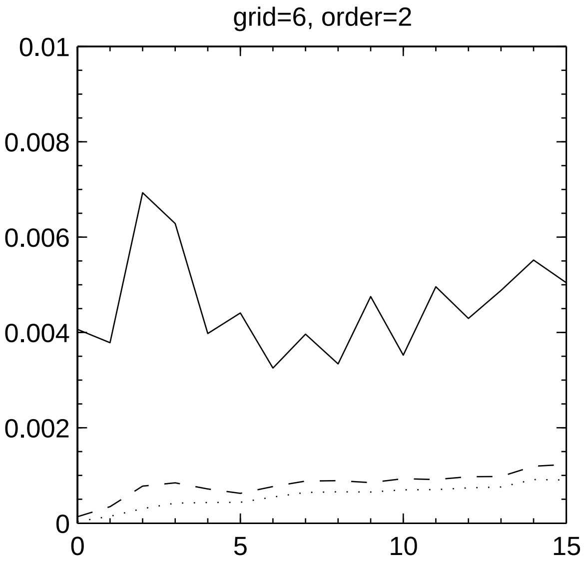
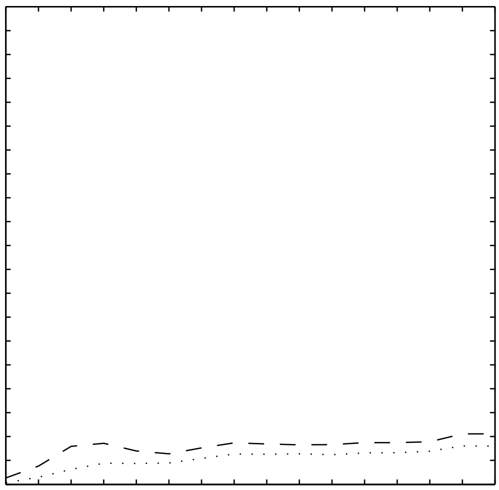


Figure 5.15: Difference between spectral and icosahedral height fields after a fifteen day run of test case 5. Contour interval is 2 m.


grid=6, order=2
0.01
0.008
0.006
0.004
0.002
0
0
5
10
15



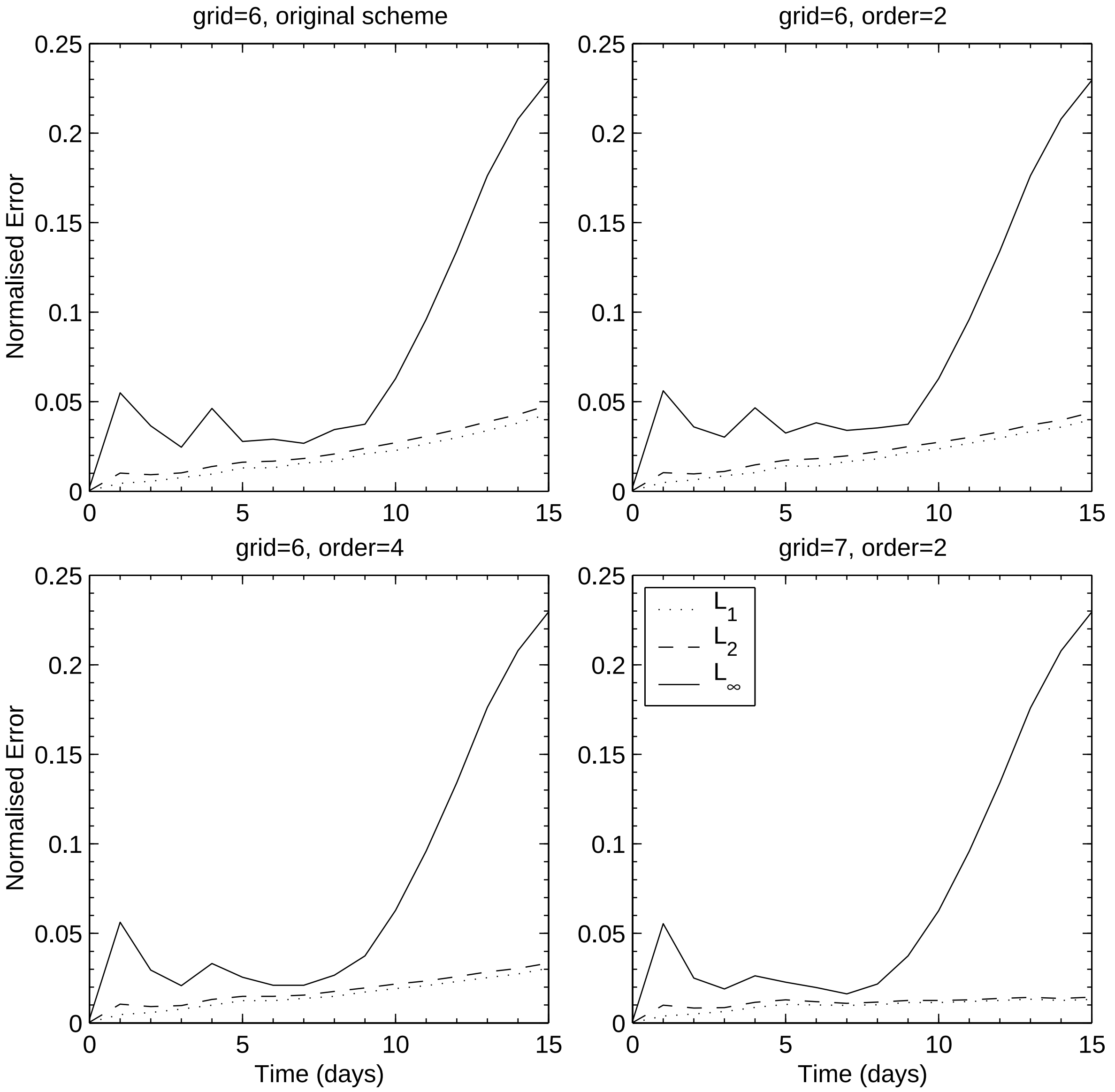
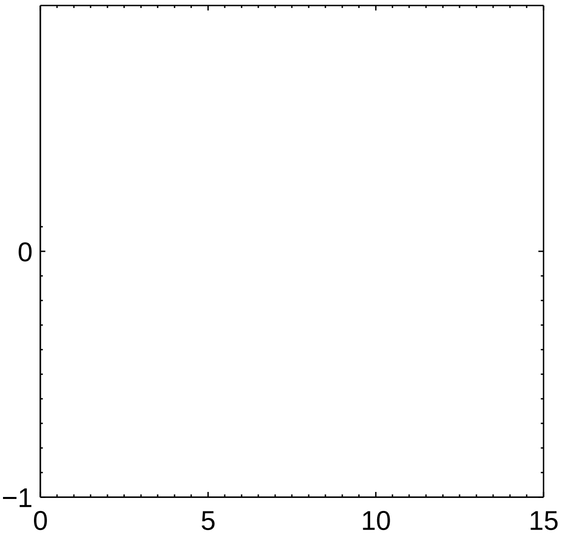


Figure 5.17: Normalised global errors in the wind field for a fifteen day run of test case 5. Errors are a comparison between numerical results and a high resolution spectral model, sampled every day.

errors in this case. The improvements in the global wind field errors as the schemes are changed, Figure 5.17, are more pronounced than for the height field. The rapid increase in the L_∞ error after day nine is the same in all cases but cannot be seen in the L_1 and L_2 errors. This suggests that the cause of this error is localised.

Figure 5.18 shows the evolution with time of the global invariants of the shallow water model. The mass δ $+$ $+$ q +gure


0
−1
0
5
10
15

does improve on these results for the other integral diagnostics.

5.7 Test Case 6:

5.7.1 Rossby-Haurwitz wave

A Rossby-Haurwitz (R-H) wave is an analytic solution of the non-divergent shallow water equations consisting of steadily propagating profile. This has become a *de facto* standard test case due to its use by many authors, despite the use of different parameters and the fact that R-H waves are not analytic solutions of the full shallow water equations.

The initial, non-divergent, velocity field is specified by the stream function

$$\psi = -a^2\omega \sin\varphi + a^2 K \cos^R\varphi \sin\varphi \cos R\lambda \ , \tag{5.30}$$

where ω, $K = 7.848 \times 10^{-6}$ s^{-1} are constants and $R = 4$ is the zonal wavenumber. The initial height is obtained from the stream function by solving a Charney balance equation so that the initial divergence tendency is zero, giving

$$gh = gh_0 + a^2 A(\varphi) + a^2 B(\varphi)\cos R\lambda + a^2 C(\varphi)\cos 2R\lambda \ , \tag{5.32}$$

where functions A, B and C are given by

$$\begin{aligned} A(\varphi) =& \frac{\omega}{2}(2\Omega+\omega)\cos^2\varphi && (5.33_A) \\ &+\frac{1}{4}K^2\cos^{2R}\varphi\left[(R+1)\cos^2\varphi+(2R^2-R-2)-2R^2\cos^{-2}\varphi\right] \ , \\ B(\varphi) =& \frac{2(\Omega+\omega)K}{(R+1)(R+2)}\cos^R\varphi\left[(R^2+2R+2)-(R+1)^2\cos^2\varphi\right] \ , \\ & && (5.33_B) \\ C(\varphi) =& \frac{1}{4}K^2\cos^{2R}\varphi\left[(R+1)\cos^2\varphi-(R+2)\right] \ . && (5.33_C) \end{aligned}$$

The background height, h_0, is taken to be 8×10^3 m. This test has significantly stronger winds than those used in other cases, so shorter time steps of 15 minutes on grid 6 and 10 minutes on grid 7 have been used.

The 'true' solution is given as the results from a high resolution spectral method as in the previous test case. In this case there are none of the problems with this solution that were seen in the case of flow over a mountain. The parameter $R = 4$ was chosen by Williamson et al. [59] who argued;

> Unstable waves [13] are not chosen, since slightly different perturbations may lead to growth of diff

5.7.2 Results

The results of the spectral model at days 1, 7 and 14 are shown in Figure 5.19. Computing this solution has not excited any of the unstable modes in the solution. When this is compared with the results after 14 days, Figure 5.20, for the icosahedral models the loss of symmetry between the waves can clearly be seen. This is more pronounced for the original scheme than for the new schemes suggesting that the original scheme introduces a larger error onto the unstable modes.

Figure 5.21 shows the difference between the icosahedral and reference solution after one day of the run. The largest errors correspond with the gradients of the wave pattern seen in the reference solution and have a wavenumber four structure. The pattern of errors is consistent across all these tests, only the magnitude varies. A closer inspection of these error fAst , 7 eows d[e H x e olus t d s c ts H b+ n ma a a a m e l fiuly H X dodes.

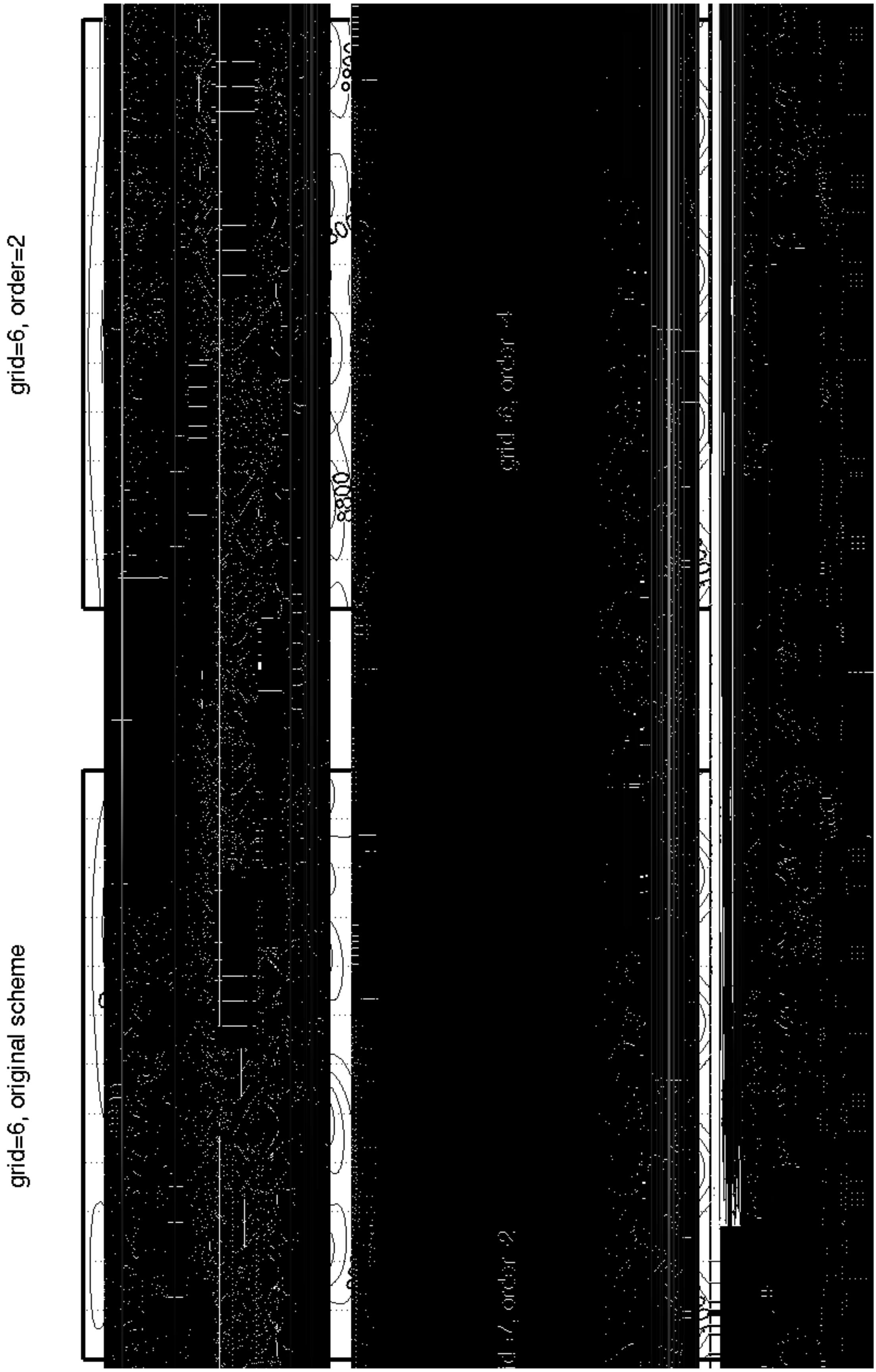


Figure 5.20: Numerical solutions for test case of a Rossby-Haurwitz wave after 14 days on icosahedral grid. Contour interval is 200 m.

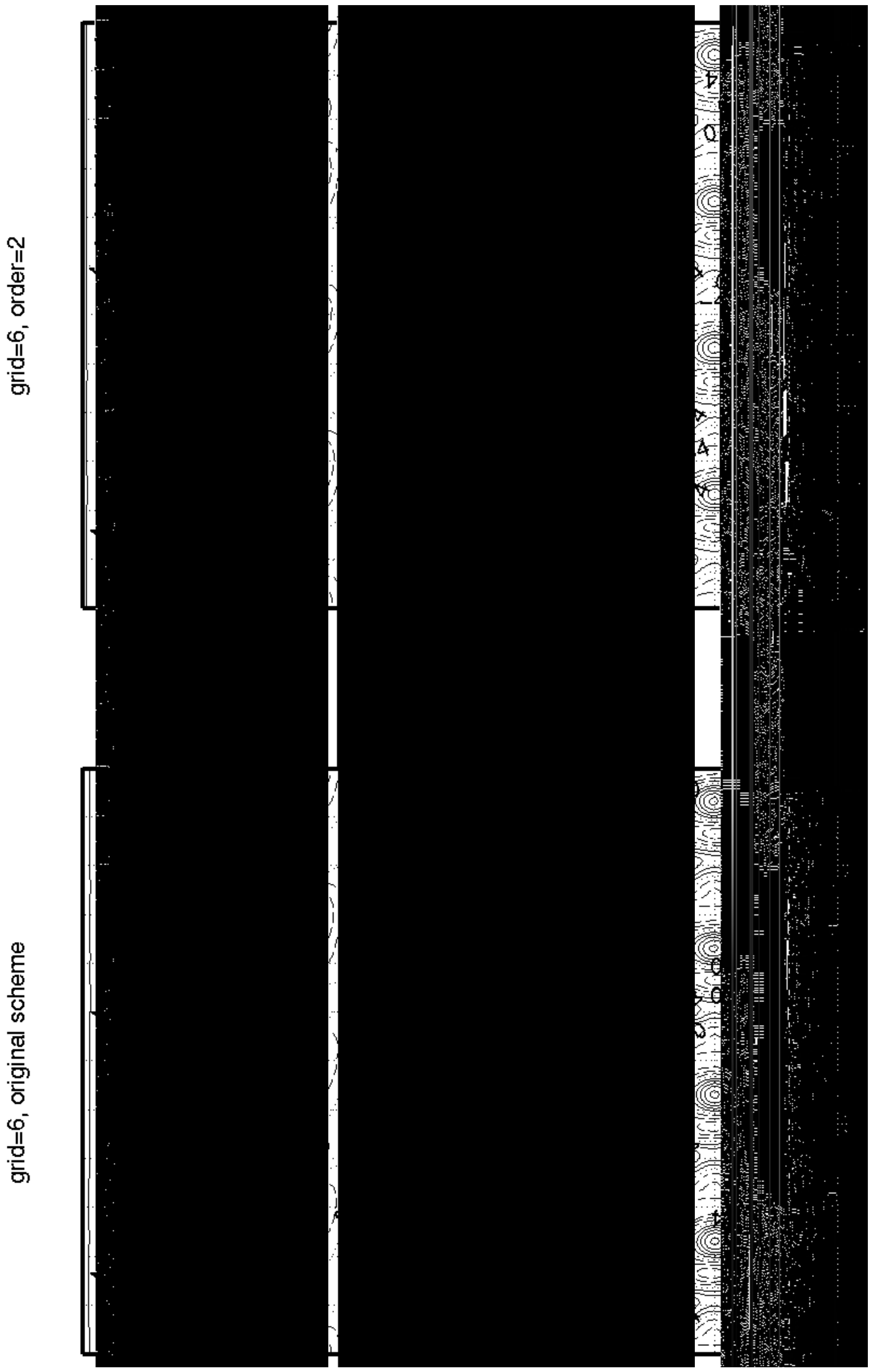
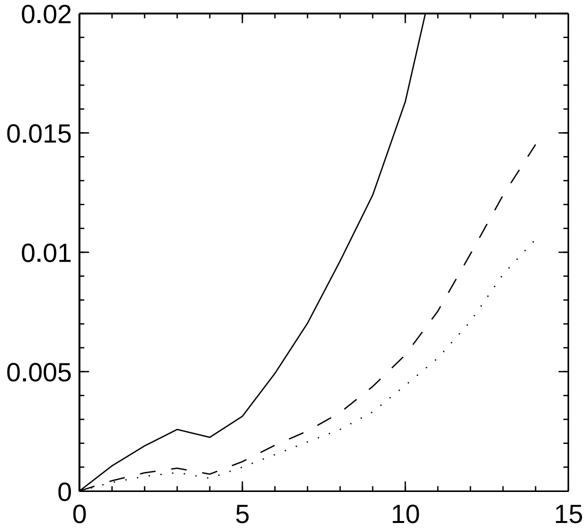
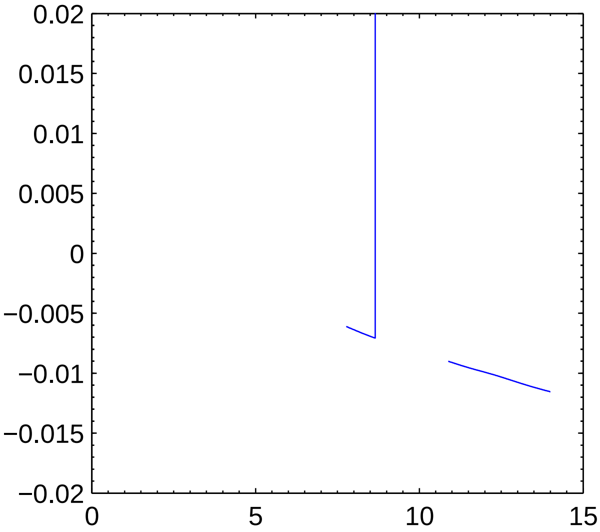


Figure 5.21: Difference between spectral and icosahedral height fields after one day of a run of test case 6. Contour interval is 2 m.


grid=6, order=2

behave as for previous cases. The global height error increases slightly for the new second order scheme over the original scheme but the new fourth order scheme is an improvement over both. As the unstable modes grow, so the error begins to grow rapidly, this happens sooner and faster for the original and second order scheme. The fourth order scheme projects a smaller error onto the unstable mode so it takes longer for this to grow and contaminate the solution, this also means that the growth is slower. Using the higher resolution grid introduces much smaller errors onto the unstable modes with the result that the global error measures show rapid growth much later in the run. This is partly due to the increased resolution and partly because of the shorter time step used in this case. The same observations can be made for the global wind errors (not shown) for this case.

The normalised global diagnostics, Figure 5.24 again show the improved conservation properties of the new schemes. The mass is again conserved by the model for all cases and is not shown. There is a greater loss of energy using the original scheme on the new grid than on the old. Increasing the resolution a 1n 1ff [illegible] p proveng t onser


0.02
0.015
0.01
0.005
0
−0.005
−0.01
−0.015
−0.02
0
5
10
15

The effect of this can be seen in the time evolution of the total vorticity in the models. This shows less fluctuation in time when the original scheme

Th lsere s hoels a δ r+cosy a sign fi ca δ sm) w 6 6

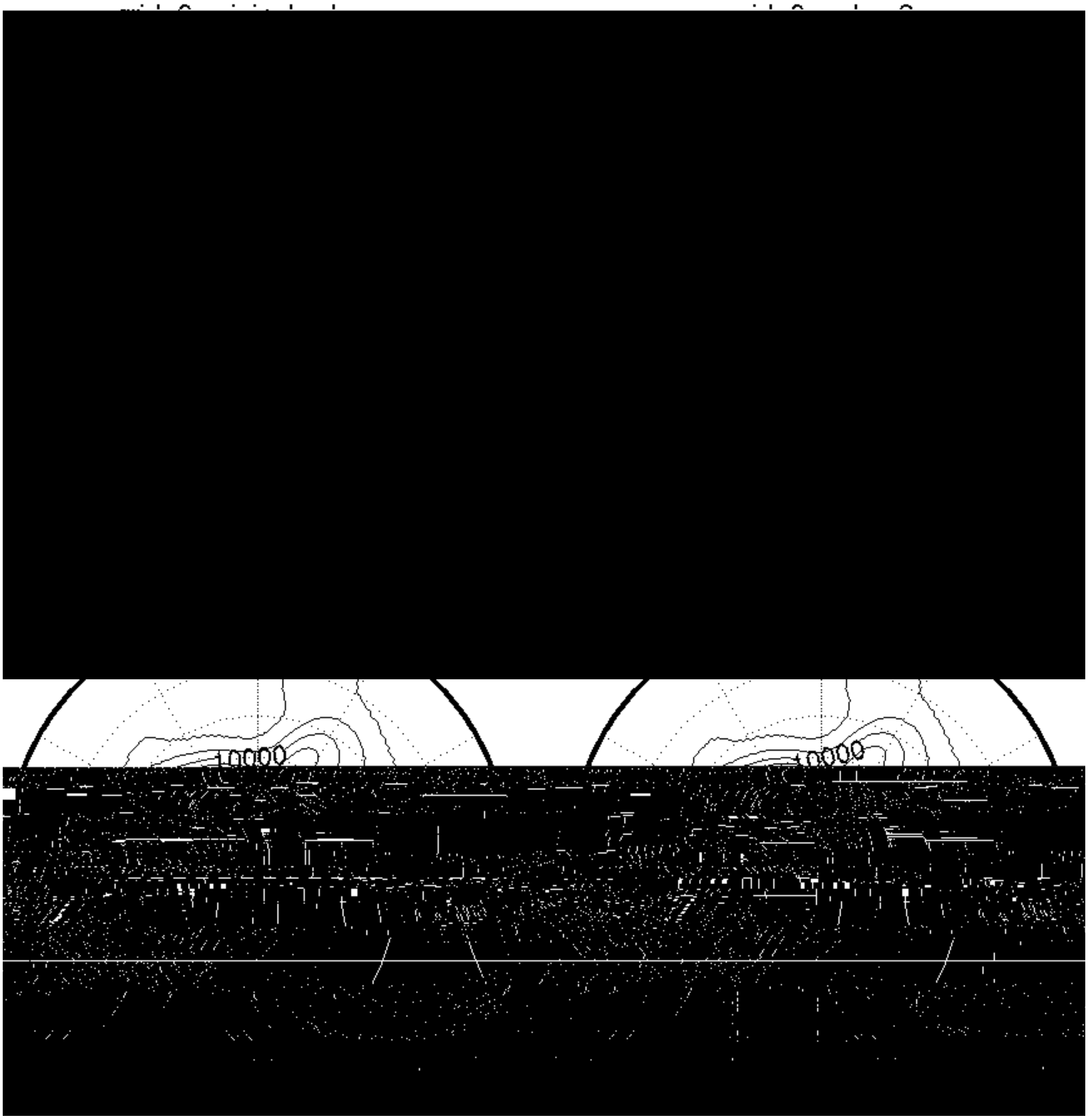


Figure 5.25: Height field in the northern hemisphere after a five day run of test case 7a. Contour interval is 100 m

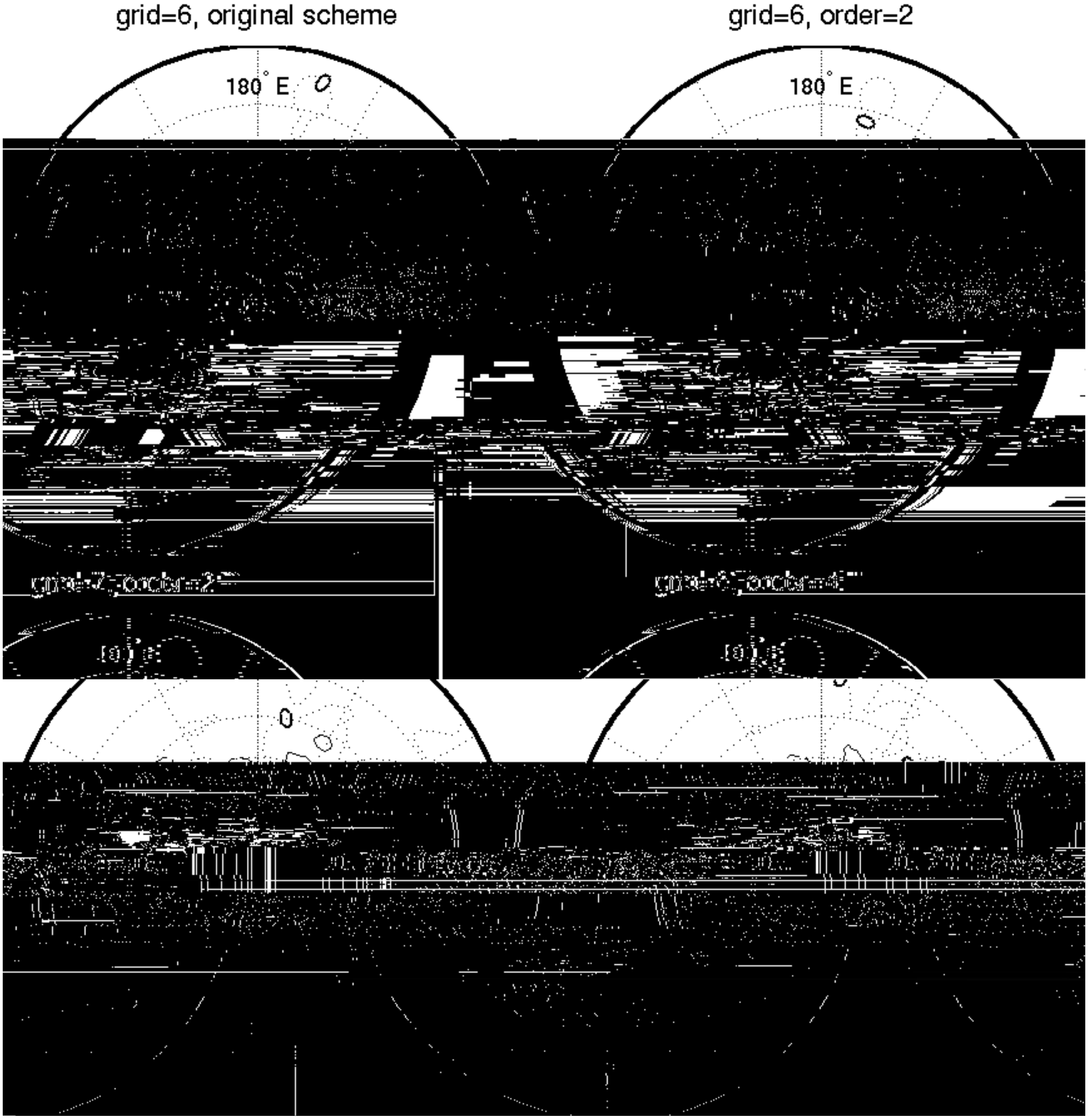
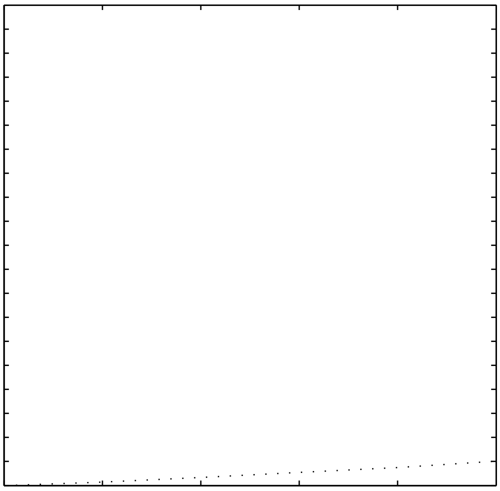


Figure 5.26: Difference between Icosahedral and spectral height fields in the northern hemisphere after a five day run of test case 7a. Contour interval is 25 m



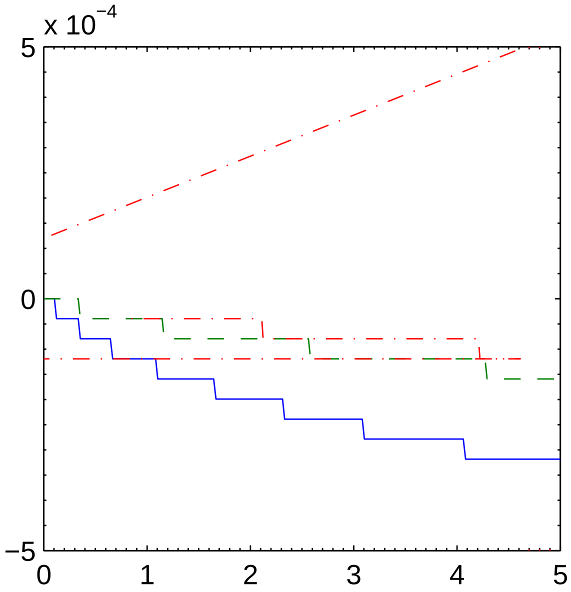
errors in the wind field (not shown).

Figure 5.28 shows the time evolution of the global diagnostics for this case, which are very similar to those for the other two initial conditions. As ever, the model conserves mass exactly to round off error. In these tests there is a slight loss of energy in all the models. The loss is greatest for the original and second order schemes on grid 6. Using a fourth order polynomial on this grid significantly improves the conservation of energy, using the higher resolution model provides a further slight improvement. The same relationship can be seen in the conservation of potential enstrophy. In this case however, it is clear that the cascade to unresolved scales is faster at the start of the run than at later times. This suggests there are a lot of small scale features in the initial conditions which are quickly lost through inaccuracies in the schemes. This process is faster for the original and second order schemes initially but the cascade to unresolved scales appears more even across all schemes later in the run.

The fields for these test cases contain an initial imbalance in the total vorticity which is maintained by the schemes used in all cases. There are some differences between the vorticity for the grid 6 schemes however, they are small compared to the magnitude of the field. The fluctuations in the total divergence is also similar for the three grid 6 models but slightly different for the grid 7 model.

The results using the fourth order advection scheme on grid 6 compare favourably with those of the T42 spectral model tested by Jakob et al. [15]. Increasing the resolution to grid 7 and using a second order scheme makes the results comparable with those of the T63 spectral model, in terms of the global error measures.

The reasons for some of the features of the results can be seen in the


x 10^{−4}
5
0
−5
0
1
2
3
4
5

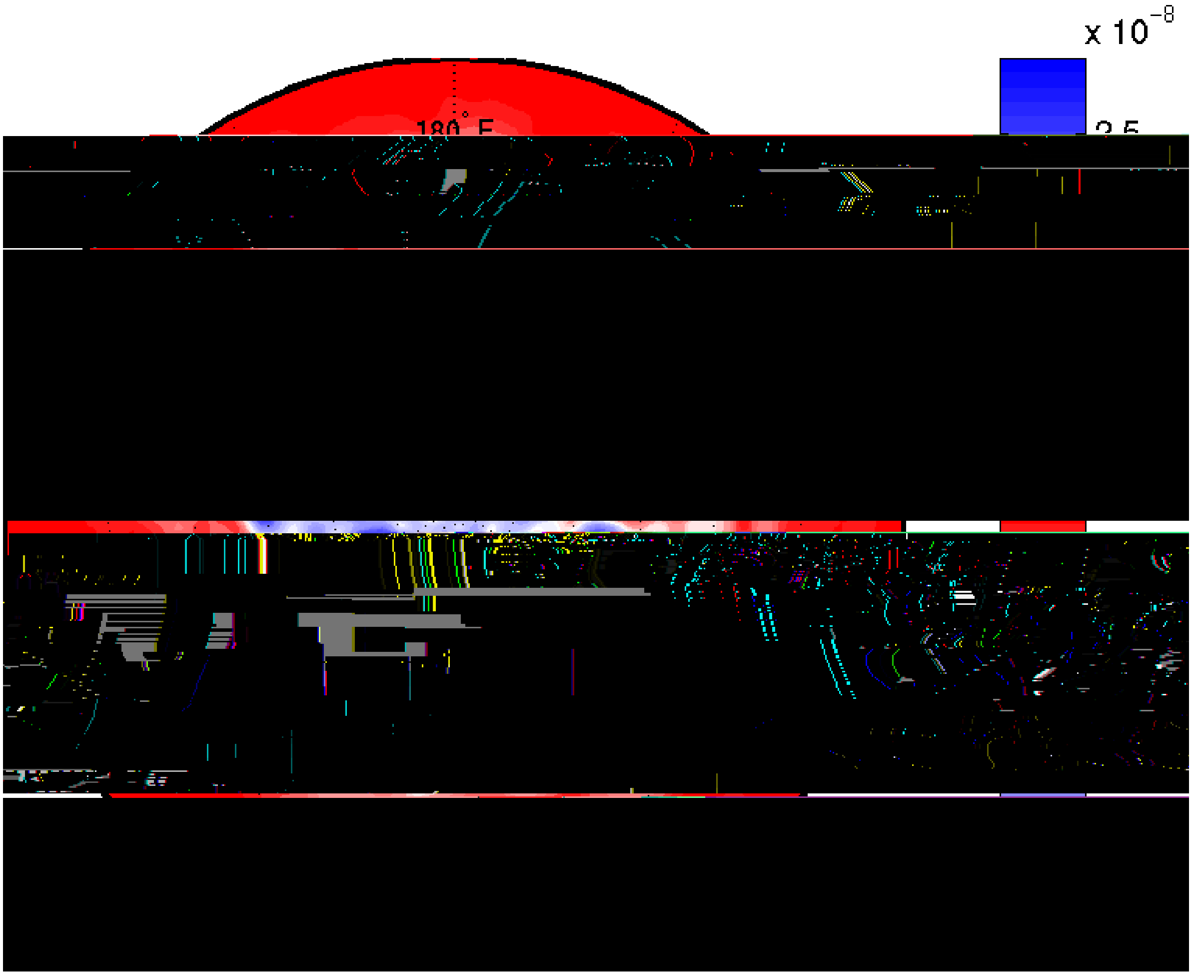


Figure 5.29: Initial PV field in the northern hemisphere for test case 7b. Contour interval is 0.25×10^{-8}, and labels are also $\times 10^{-8}$

PV fields of the models. The initial PV field is shown in Figure 5.29. This contains a lot of small scale features and “steep” gradients in the field. These features will be dissipated faster by the original and second order schemes on grid 6 which will cause the faster cascade of potential enstrophy to unresolved scales. This was seen in the global diagnostics and can also be seen in the PV fields after the first day of the run, Figure 5.30.

Comparing the PV fields of the models after five days, shown in Figure 5.31, shows the greater improvements in the higher order and higher resolution tests. Much of the detail that was still present after one day

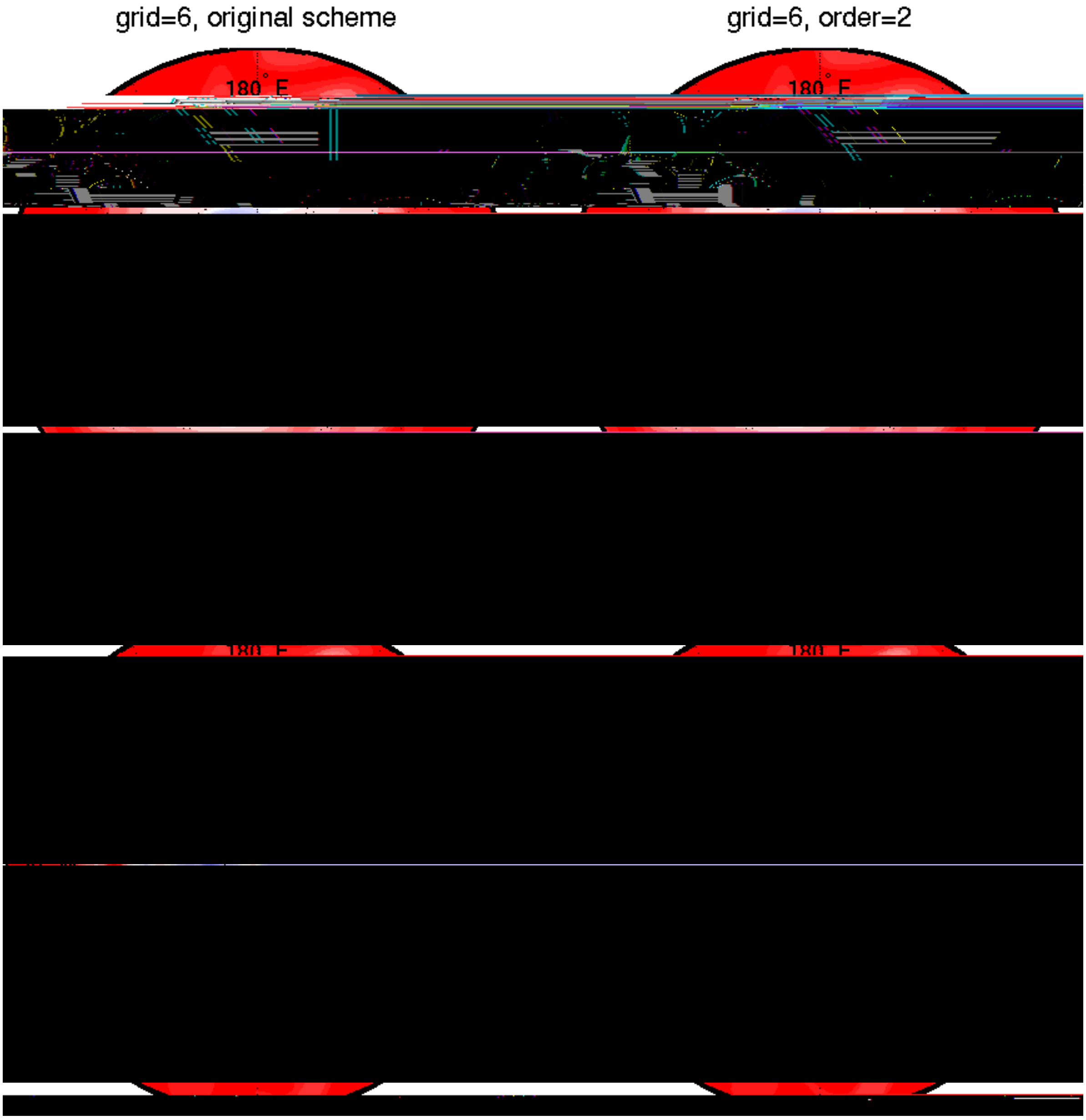


Figure 5.3

has

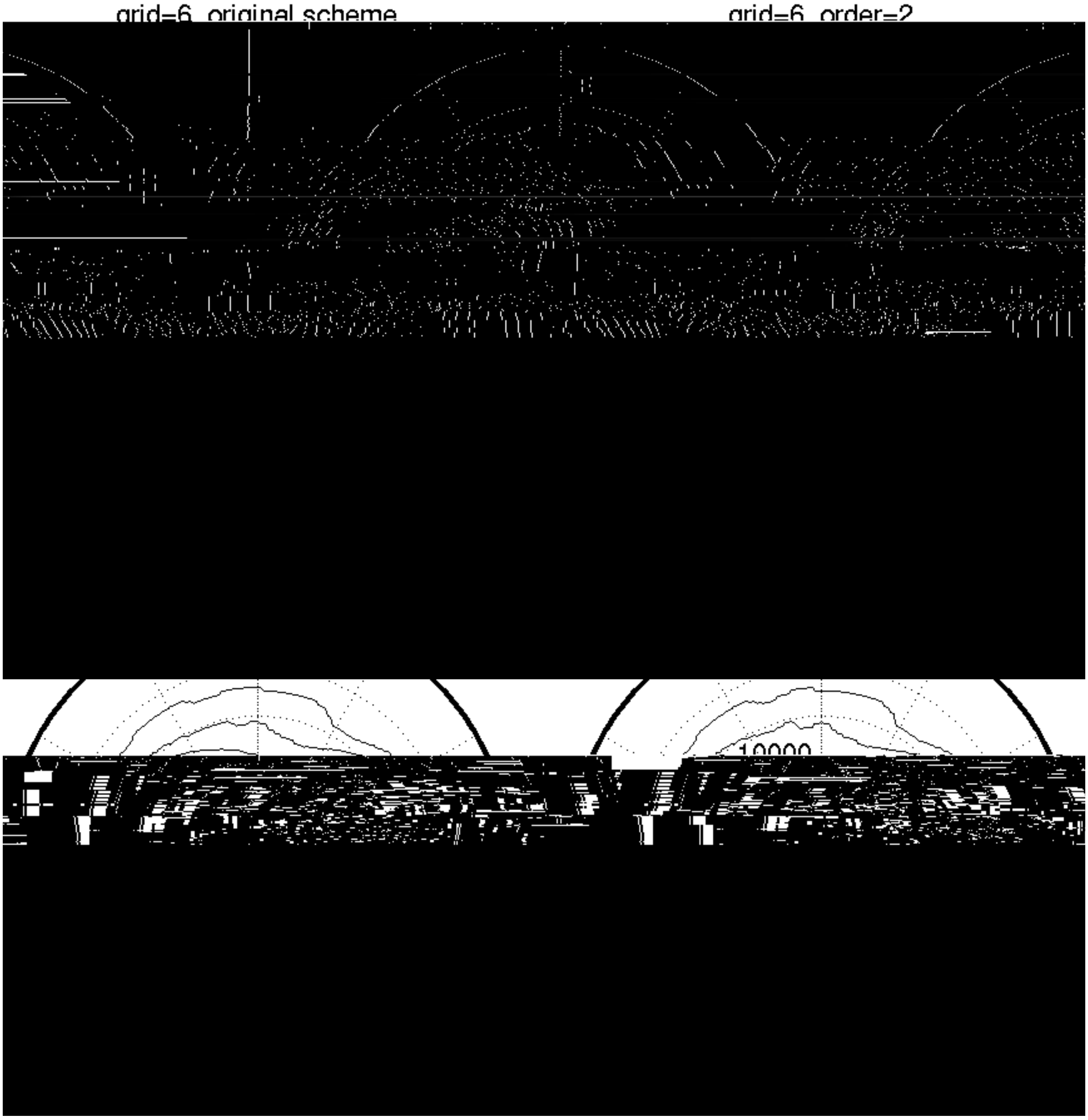


Figure 5.32: Height field in the northern hemisphere after a five day run of test case 7b. Contour interval is 100 m

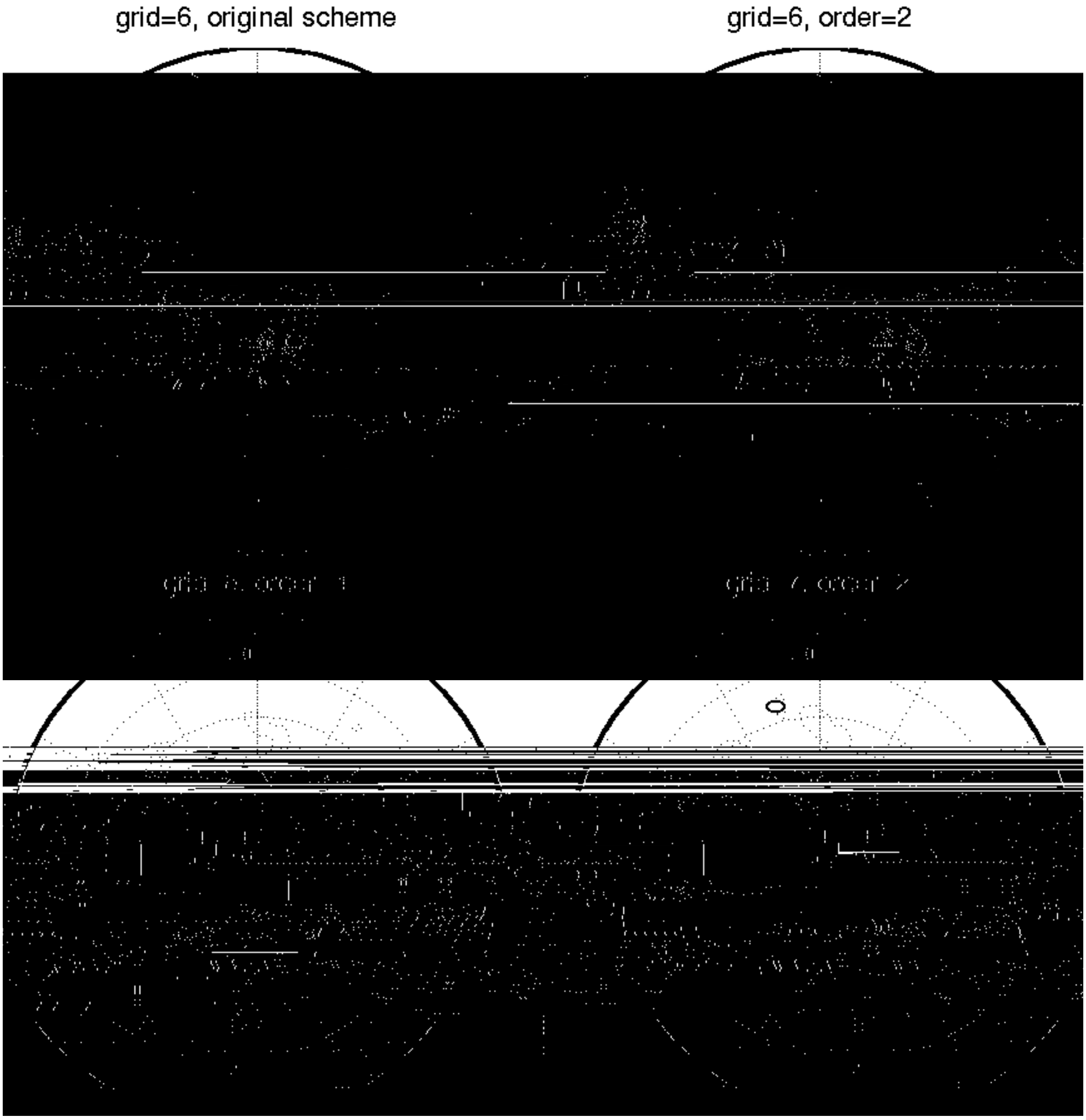
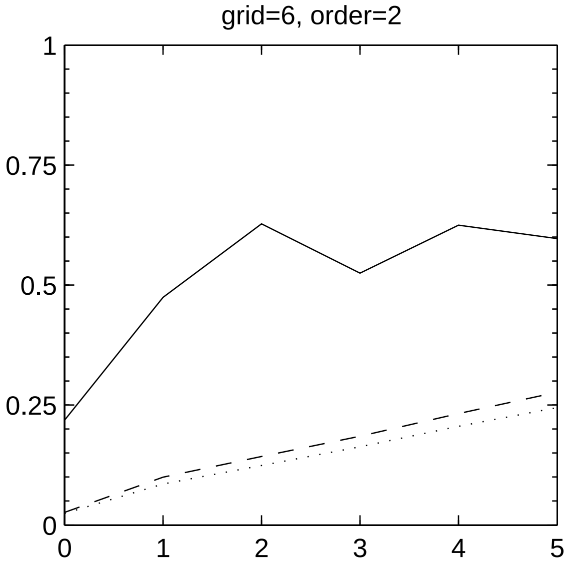
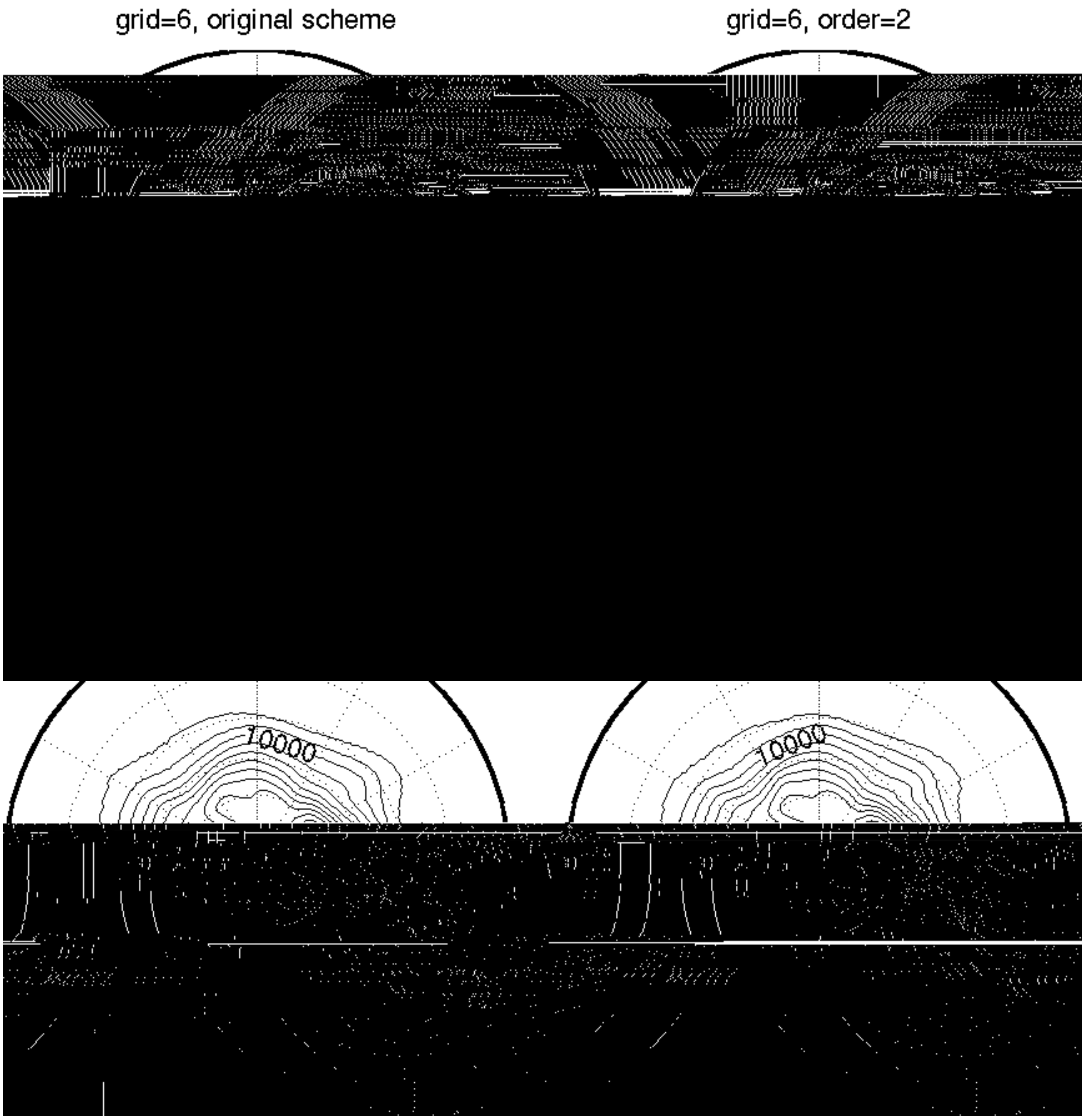


Figure 5.33: Difference between Icosahedral and spectral height fields in the northern hemisphere after a five day run of test case 7b. Contour interval is 25 m


grid=6, order=2
1
0.75
0.5
0.25
0
0
1
2
3
4
5

the T



grid=6, original scheme

grid=6, order=2

Figure 5.35: Height field in the northern hemisphere after a five day run of test case 7c. Contour interval is 100 m

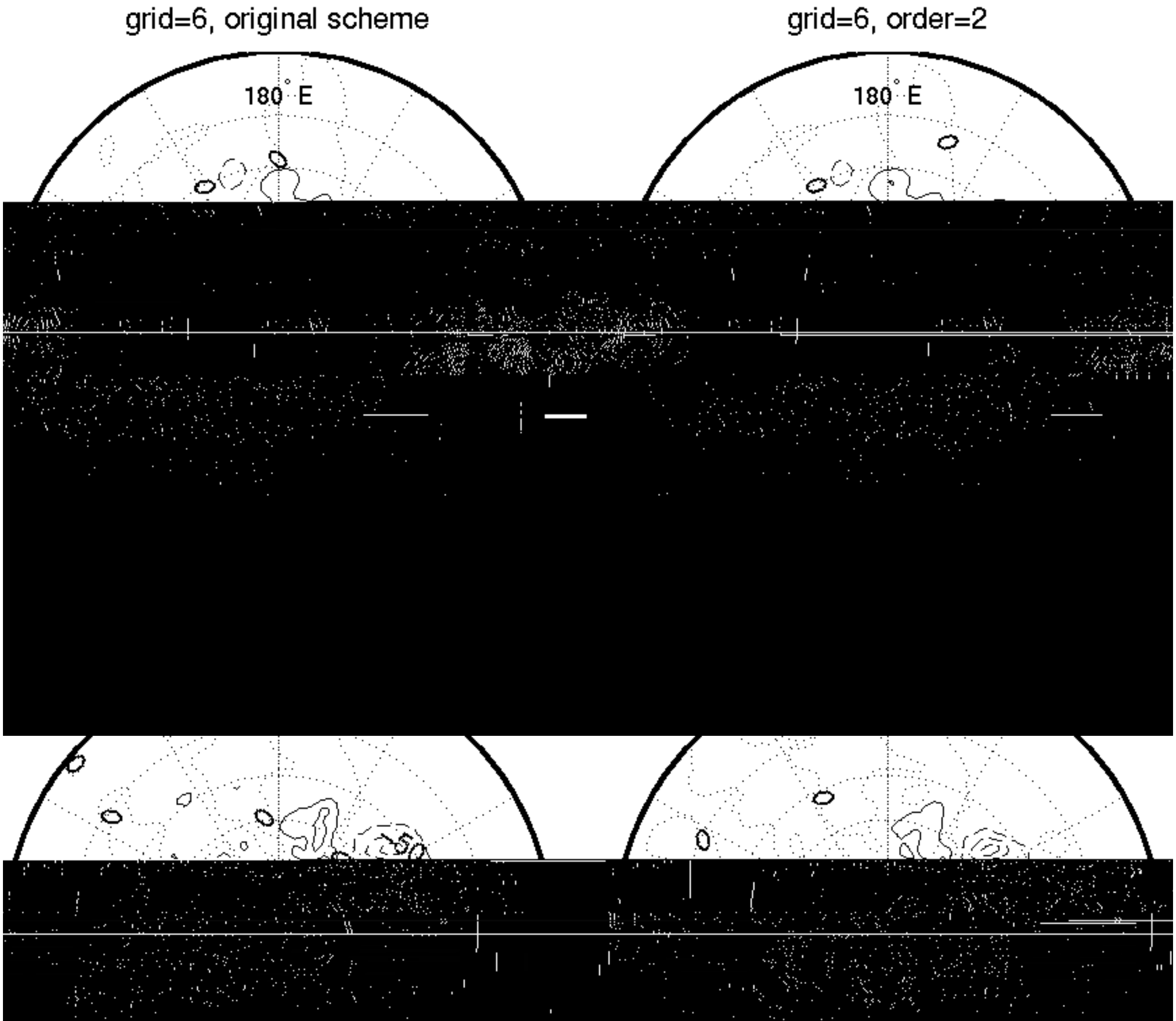


Figure 5.36: Difference between Icosahedral and spectral height fields in the northern hemisphere after a

compared with both the original model and other published results (the icosahedral model of Heikes and Randall [12] and the spectral model of Jakob et al. [15]). The model used here used the tweaked icosahedral grid whereas the un-tweaked grid was used by Thuburn in [52]. Tests using the original advection scheme with the tweaked grid were used to distinguish between changes due to the grid and those due to the advection scheme.

The only significant differences between the results when using the original scheme on the two grids occured in tests two and three. These test cases both involved steady states of the equations, so changes to the grid changed the discrete steady state and the accuracy of the approximation to

$$\frac{\partial ()}{\partial t} = 0 \ , \tag{5.34}$$

Changing the advection schemes used within the model made little difference to the overall errors but did change their spatial distribution. This suggests that the advection scheme used is not the main cause of the errors in these cases but does have some effect on the results.

In general, the original scheme and the new second order scheme performed in a similar fashion for all the tests. Both these schemes are based on the fitting of a second order polynomial over a similar region upwind of each edge. The similarity of the results suggests that the method of the fitting has little effect on the overall results of the method.

Using the fourth order scheme instead of the second order one made a larger improvement in the results of some tests than in others. In test four for example, the fourth order scheme prevented any significant filling of the low centre. In test case five however, the only significant improvement was in the region downwind of the mountain, a region dominated by advection. The improvements in test case seven appear to be small when looking at the global errors yet there are significant improvements to the 'forecast' when looking at specific regions of the model.

Increasing the resolution of the model had a greater improvement on the results than increasing the order of the advection scheme. This came at an increased computational cost as roughly four times as many cells and edges were used on grid 7 than on grid 6. In test cases two and three there was a large improvement in the results for this increase in computational cost but this was not always the case. Whilst the global errors for test case 7 were improved over the fourth order scheme on the coarser grid, the improvements were small. This suggests that the most cost effective improvement to the second order scheme on any grid is to increase the order of the scheme used. If better accuracy is all that is required then the grid should be refined and the computational price paid.

The results from the test seven cases have shown that the higher order advection schemes generated by the new method have some advantages over the lower order schemes. This is despite the global errors showing only a small improvement in many of the earlier tests. The ability of the higher order schemes to capture smaller scale features of the flow will have little effect on the results in some regions and more in others.

In the real atmosphere there are some regions that are more sensitive to perturbations in a field than others. These perturbations may be caused by any of the physical and numerical processes that are used in forecasting or, in a numerical model, by the numerical methods used. Improving the numerical methods used in these regions can make a big difference to a particular forecast for a particular region. Though we do not generally know where these regions are, impro $_c$ Ten

Chapter 6

Summary and Further Work

6.1 Summary

The aim of this work was to develop accurate numerical approximations to the advection equation for use on arbitrary grids. The prime reason for wanting to do this was to improve the accuracy of geophysical models using the icosahedral-hexagonal grid on the sphere. We therefore wanted to ensure that the scheme was conservative and shape preserving as well as accurate and computationally efficient. The nature of the grids necessitated that the method was multidimensional and could be generated automatically for each grid. Using a finite volume scheme and approximating the fluxes across cell boundaries guaranteed conservation. Applying a flux limiter to the fluxes ensured shape preservation but had an adverse effect on other measures of the accuracy of

On an irregular grid this calculation is much more complicated on paper but can be easily handled by computer code. The numerical results produced $\mathbf{p}$ 1 schemes that best balance the accurate approximation of the advected field with the need to use the limiter to ensure shape preservation.

Trying to apply the same method of generating advection schemes to triangular grids provided one significant new problem, selecting a stencil over which to fit a polynomial. The problem is that there are some grids for which there is no obvious way to choose between two cells for inclusion in the stencil without considering the flow field. Using the flow field in this way is not a satisfactory solution to the problem since it requires that a computationally expensive part of the algorithm be recomputed whenever the flow field changes. Instead we select a stencil with more than the required number of cells and find a 'best fit' polynomial over this stencil. The best fit is found using a weighted least squares cost function where the weights are chosen to favour the fit over the central cell in the stencil.

The advection schemes generated by this method show a similar relationship between computational cost and accuracy as was seen on the rectangular grids. The trend is for an improvement in accuracy as the order of the polynomial or the resolution is increased. The results again suggest that a polynomial between second and fourth order should be used for maximum efficiency when the limiter is used. The limiter also causes some anisotropic distortion, dependent on the flow angle relative to the grid in this case. This was the only grid for which this was observed.

The same method was then applied to the icosahedral-hexagonal grid on the sphere. The only difficulty in this case was that the grid was spherical whilst the polynomials and integrations had previously been formed on two dimensional planes. This problem was overcome by projecting the region around each cell onto a plane for the polynomial interpolation and approximation of the fluxes. The results of test cases using schemes generated by this method behaved in the same way as on the rectangular and triangular grids. The general trend was for accuracy to improve as the polynomial order or resolution were increased. For best efficiency, comparing the computational cost and accuracy, schemes of between second and fourth order should be used.

The final method of generating advection schemes follows the following pattern;

(i). **Select a stencil:** Begin with one cell and add all cells neighbouring those in the stencil until there are more cells than there are terms in polynomial that is to be 'fitted'.

(ii). **Set up polynomial matrix:** Integrate a general polynomial over each cell in the stencil to get $\mathsf{B}\mathbf{a}$ where $\mathbf{a}$ is the vector of the coefficients of the polynomial. A (v). **Ca**

shallow water model showed that many of the local improvements in the higher resolution model could be gained by using a higher order advection scheme on a coarser grid.

The greatest strength of this method is probably speed up the overall method but would probably also have a detrimental effect on the accuracy. Fewer, bigger bins would mean a greater speed-up but less accuracy. If storage were not a problem then maximum speedup could be gained by storing the coefficients for each bin rather than recalculating them when needed. The relationship between the size of the bins and the accuracy would need to be investigated by running similar tests to those in previous chapters.

Another way of improving the efficiency of the method would be to only use the higher order schemes in regions where this will make a significant improvement in the results. These regions are likely to be where the field is less smooth and a more accurate interpolation of the field will significantly reduce diffusion. By including a simple switch into the model so that higher order schemes are used in regions with steep gradients, much of the accuracy of the higher order scheme can be gained without all the added computational cost. This technique is known as p-type refinement.

A similar idea could be used when several different fields need to be advected. Relatively smooth fields (e.g.height) could use a lower order advection scheme than fields that contain more sharp gradients (e.g.PV). In terms of the shallow water model used in this work we could use a second order scheme for the smooth height field and a fourth order scheme for the PV field. This would require more initial computation and storage but would speed up the time stepping over the fourth order scheme. The loss of accuracy caused would need investigating, again by using similar tests to before.

The accuracy of the schemes depends not only on producing an accurate reconstruction of the advected field but also on the approximation

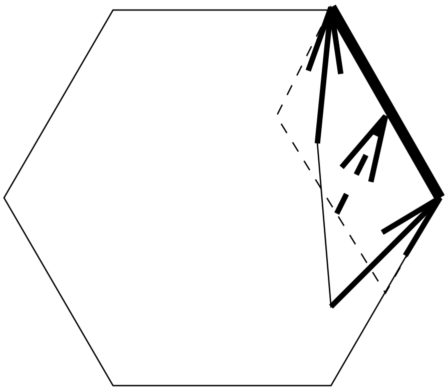


Figure 6.1: The region swept across each edge. The dotted region shows the current approximation to this region. The solid region gives a more accurate representation of this region.

of the region swept across each edge. The spatial interpolations of the advected field use high order polynomials to increase accuracy. The region swept across the edge is approximated simply as a projection of the vertices of the edge in an upwind direction. The direction and magnitude of the wind are found at the centre of each edge by a second order approximation to the TTh^T TTlnd l TTle- lT Tg l TT^T uce lar phenomenon or investigations of new theories and ideas, e.g. a cloud model or the reformulation of equations into new variables. The advantage of this scheme in this case is that the accuracy can be easily improved by using a higher order polynomial. A particular accuracy may be required to capture a certain feature but this may not be known in advance. This method of generating advection schemes allows greater accuracy to be achieved without the need to increase the grid resolution. The overall improvement in accuracy will depend on the relative importance of advective processes in the model. Each model will behave differently as the advection scheme is changed or the resolution is increased, which models will benefit from using this method is difficult to tell in advance. Performing tests comparing accuracy and computational cost on such a model may help to explain this behaviour.

This method of generating advection schemes may also be of particular use in oceanography. Oceans pose two particular problems that are not present in the atmosphere. The first of these is the domain over which the modelling is performed, which not only contains horizontal boundaries but ones that are highly irregular. Using an irregular grid over the domain is a technique that is not widely used but one which this method could easily cope with. The other problem is that the dynamics of the oceans generate strong currents along the western boundaries of ocean basins. These are very important features but pose difficulties for modelling because of their dimension and the steep gradients they cause in some model variables. This method has two benefits in this case, the first is its ability to capture sharper gradients using higher order polynomials. The second benefit is again its ability to handle irregular grids which may be useful for providing increased resolution.

Some of the ideas used with this method could be used to develop a conservativ ł TTh I TTTh I TTh ł - T m m LT T m

Bibliography

[1] J. R. Bates, Y. Li, A. Brandt, S. F. McCormick, and J. Ruge. A global shallow-water numerical model based on the semi-Lagrangian advection of potential vorticity. *Quaterly Journal of the Royal Meteorological Society*, 121:181–2005, 1995.

[2] J. P. Boris and D. L. Book. Flux corrected transport. I: SHA

[8] A. Harten, B. Engquist, S. Osher, and S. R. Ch

[16] H. O. Kreiss and J. Olger. Comparison of accurate methods for the integrat

[24] R. P. Leonard, M. K. MacVean, and A. P. Lock. The flux-integral method for multidimensional convection and diffusion. *Applied mathematical modelling*, 19(6):71–100, 1995.

[25] R. J. LeVeque. *Numerical Methods for Conservation Laws*, chapter 16, pages 173–192. Lectures in Mathematics - ETH Zürich. Birkhäuser Verlag, 1992.

[26] P. Lynch. Richardson's barotropic forecast: A reappraisal. *Bulletin of the American Meteorological Society*, 73:35–47, 1992.

[27] B. Machenhauer. Spectral methods. In *N*

[31] A. McDonald. Semi-lagrangian methods. In *Numerical methods in atmospheric models*, pages 257–271. ECMWF, Sept 1991.

[32] M. E. McIntyre. Atmospheric dynamics: Some fundamentals with

hexagonal grid for the sphere. *Monthly Weather Review*, 96:351–356, 1968.

[40] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hy

the shallow water equations in spherical geometry. *Journal of Computational Physics*, 102:211–224, 1992.

[60] M. A. Wlasak. *The examination of balanced and unbalanced flow using potential vorticity in atmospheric modelling.* PhD thesis, University of Reading, 2002.

[61] S. T. Zalesak. Fully multidimensional flux corrected transport algorithms for fluids. *Journal of Computational Physics*, 31:335–362, 1979.