

UNIVERSITY OF READING

The Shallow-Water Semi-Geostrophic Equations
on the Sphere

Mark H. Mawson

A thesis submitted for the degree of Doctor of Philosophy.

Department of Mathematics

August 1994

Abstract

The semi-geostrophic equations of meteorology are considered in their shallow-water form on the sphere. To distinguish between the equations considered here and those described by Shutts and Salmon we refer to our equations as the “spherical semi-geostrophic equations”. It is shown that there may exist a unique solution to these equations on the sphere for finite time under given conditions if certain key results can be proven. A robust numerical method is developed based on a finite-difference predictor/corrector scheme which uses a multigrid algorithm to solve the elliptic equation which arises in the correction step. Numerical techniques are developed which restore numerical solutions which at any stage fall outside the solution set for the equations back into it. Eulerian and semi-Lagrangian prediction schemes are compared for simulations starting from both idealised data and data taken from an operational numerical weather prediction model. It is found that the semi-Lagrangian scheme is superior in the vicinity of the pole, since it does not generate small-scale numerical noise and better preserves the shape of features. The results for the spherical semi-geostrophic equations are also compared with those from a shallow-water primitive equation model. Clean comparisons for idealised data simulations are difficult to obtain due to the scarcity of analytic solutions on the sphere which are common to both equation sets. Solutions for real data simulations are also presented for both spherical semi-geostrophic and primitive equation shallow-water models. They are found to be in good agreement for the large scale features present in the real atmospheric data.

Acknowledgements

The first, and largest, acknowledgement goes to Dr Mike Cullen for his contributions to my understanding of this work and his wealth of knowledge in both analytic and numerical techniques, and most importantly, for introducing me to this area of work in the first place. Next to Dr Mike Baines, my supervisor at Reading University, as the hour spent each Friday morning explaining things to him was usually the most useful of the week. Also for his many helpful contributions to the work, including finding all the little things I sometimes overlooked. To Dr Ray Bates for kindly supplying the semi-Lagrangian primitive equation code. To Dr Gareth Shaw for supplying the code which formed the basis of the multigrid code and for answering my questions about multigrid methods. Last, but not least, to Dr Andy Priestley of Reading University for his many helpful comments.

I would like to acknowledge that this work could not have been done without the financial support of the U. K. Meteorological Office, nor without their willingness to allow their staff to pursue research qualifications as part of their full-time job.

Contents

1	Introduction	1
1.1	The semi-geostrophic equations.	6
1.2	Other comparable balanced equation sets.	12
2	Theoretical results	15
2.1	Known results for the f -plane case	16
2.1.1	Existence and Uniqueness on a f -plane	16
2.1.2	Application of this Procedure to the Sphere	19
2.2	Results for the equations on the sphere	21
2.2.1	SSG Energy Conservation	21
2.2.2	SSG Absolute Vorticity Evolution Equation	23
2.2.3	Existence and uniqueness of solutions on the sphere	26
2.2.4	Comparison of this Method with other methods for balanced equations	38
2.3	Solutions beyond the breakdown	40
2.4	The Analytic approach as a Numerical Method.	43
3	Numerical solution procedure	45
3.1	The Predictor/Corrector Approach	46
3.1.1	The Advection Problem	47

3.1.2	The Correction Problem	49
3.1.3	Solving the system by Advection then Correction	52
3.1.4	Convergence Results derived from Numerical Experiments	55
3.2	Numerical Implementation of the Predictor/Corrector Method . .	59
3.2.1	Computational Grid Choices	60
3.2.2	Prediction Schemes	67
3.2.3	Constraint Enforcement	82
3.2.4	Correction Step	87
3.2.5	The Complete Procedure	94
3.3	Multigrid Methods on the Sphere.	95
3.3.1	Introduction to Multigrid methods.	95
3.3.2	Solution of a class of second-order P.D.E.s on the sphere.	98
3.3.3	Implementation	100
3.3.4	Results.	110
3.3.5	Discussion of future developments and parallelisation as- pects.	120
4	Computational Results	124
4.1	Data Initialisation	124
4.2	Idealised Data	128
4.2.1	McDonald-Bates 1989 Problem	128
4.2.2	The Rossby-Haurwitz Problem	146
4.3	Real 500hpa Height Data	154
5	Conclusions and Discussion	168
5.1	Conclusions	168
5.2	Discussion	170
5.2.1	Extension to 3 dimensions	170
5.2.2	Inclusion of friction	171

Chapter 1

Introduction

In atmospheric modelling we are interested in predicting the motion and properties of a fluid on a sphere. For the Earth's atmosphere this is a very complicated problem with many microphysical processes interacting with the dynamical fluid motion, for example latent heating due to the changes in phase of water in the atmosphere. The system is also subject to external forcing from upper and lower boundaries. For example, at the lower boundary there are the fluxes of heat and moisture into and out of the oceans and land masses, whilst at the upper boundary there is radiative forcing. Added to these effects are the more unpredictable ones such as those caused by volcanic eruptions or other natural phenomena. These unpredictable elements are beyond our control and we seek to provide the best possible model of the atmosphere for use in both operational weather forecasting and long time climate simulations. The models in current use are based on a simplified form of the Navier-Stokes equations, where approximations appropriate to the scales resolvable by current computing power and to the nature of the atmosphere have been made. These equations are supplemented by many schemes, called *parameterization* schemes, which seek to represent the effects of the microphysical processes and the unresolved motions on the large scale quantities predicted by the equations. The first person to realise that this was possible

was V. Bjerknes in the early part of this century, who recognised that there was a deterministic set of equations for the atmosphere. He also showed that the system was highly non-linear and that in general there was no analytic solution. Undaunted by this L. F. Richardson performed the first numerical weather prediction experiment, publishing his results in 1922, [59]. The calculation took several months, and unfortunately the results were found to be in error by several orders of magnitude. Perhaps because of this error, or more likely because of the time and labour involved in obtaining the results, numerical prediction remained only a hypothetical possibility until the advent of the computer. During this time the forecasters continued to use their tried and trusted methods, many of which are still commonly used today in the forecasting of phenomena, such as fog, which numerical models have yet to predict well. The techniques used by forecasters during the early part of this century are described in Shaw [67]. The first successful numerical prediction came not long after computers first appeared and was due to Charney, Fjortoft and von Neumann in 1950, [12]. Since then the rapid advances made in the power of computers have led to equally rapid growth in the complexity and size of the atmospheric models run on them. A typical operational global model in use today has a horizontal resolution of 100 kilometres and a vertical resolution of 1 kilometre, this requires around 1.2 million data points per variable held. The resolution of the model is constrained by the power of the computer and how quickly you wish to obtain the forecast from the model. To provide more detailed forecasts of what is commonly considered as weather, namely, cloud, rainfall, fog, and wind, for a particular area of the world a model with higher resolution than the global one is run for that area. Such models typically have horizontal resolutions of 30-50 kilometres and better vertical resolution in the lowest 10 kilometres of the atmosphere, of the order of 500 metres. For climate simulations, which need to be run for many decades to obtain statistically significant results, lower resolution global models are used as

they require less computer time. The complexity of the atmospheric models has increased substantially over the last 20 years, particularly the representation of the microphysics in the numerical model. These increases in complexity led to better forecast and climate predictions. However it was also possible to obtain significant increases in forecast skill simply by running the models at higher resolution, and such improvements were generally obtainable every five years with the arrival of a new, faster, generation of computer. The experience of many numerical weather prediction centres is that this situation has now ceased to be true and much work is required when increasing the resolution to obtain any significant improvement. One reason for this is that the resolution of models has now reached the point where the separation into resolved and unresolved motions for certain atmospheric phenomena is less clear cut, see Smagorinsky [70] for a figure showing the time and space scales of atmospheric motions. The interaction between the motions on the various scales is important in the correct modelling of such features, and thus the interaction between the parameterization schemes and the large scale equations must be well understood. It is now likely that every time we change the model resolution we must spend considerable time understanding the interactions at that scale in order to obtain a better forecast. To do this, we must gain understanding about these interaction processes and how we are modelling them. Another limiting factor is our knowledge of the initial conditions. The worldwide observing system is able to provide only a small fraction of the data required to initialise the model at any one time. We have to produce a best estimate of the initial conditions for the model based on the available data and that data which has gone before, plus information from the previous run of the forecast model. The problem is discussed in detail in Daley [24]. Operational forecast models can be very sensitive to small perturbations in the initial conditions, and this is another area where more understanding is required to make the best use of the data we have and in trying to determine which is the most likely

scenario.

In order to improve the current numerical weather prediction models it is essential that we understand more about the real atmosphere and more about our numerical model as well. In particular, we need to understand how the model reacts to the forcing we are providing. To gain insight into how both the atmosphere and the model work much work has been done with simpler models of the atmosphere. These simpler models are usually based on equations which have had a further level of approximation made to them, and in the early days of computers were used as the basis for providing weather predictions, for example Charney et al. [12]. Much work has been done with these simpler equations confined to a plane where the Coriolis force due to the Earth's rotation is assumed constant, known as the f -plane approximation. This approximation is valid for many phenomena of interest in mid-latitudes and much insight has been gained from running these models in this way. We wish to discover if more insight can be gained into more general phenomena by considering one of these simpler models on the sphere. In particular we would like to be able to run such a model using real data and compare the results with those gained from an operational model. This, it is hoped, will allow us to investigate many questions of interest, such as the importance of representing fast timescale motions explicitly rather than as a parameterization. In this thesis we take one of these equation sets and extend it to the sphere. The equations we choose to study are essentially the semi-geostrophic equations first suggested by Eliassen [27] in 1948, the usefulness of which was first really seen in the work of Hoskins and Bretherton [38] in 1972. In their paper they demonstrated the ability of the equations forced by a deformation field to produce idealised atmospheric fronts which possessed many of the characteristics of real fronts. Since then these equations have been used extensively to investigate, in particular, the role of moist processes, such as latent heat release and evaporation, in the formation of atmospheric weather

fronts, see for example Holt [34], Holt and Thorpe [35], Castelli et al. [10], to list just a few. They have also been used to simulate idealised flow over a mountain, Cullen et al. [17], Pierrehumbert [58], where the topic of interest was the effect of a barrier on a uniform flow normal to it. This is of interest in the simulation of downslope wind storms where sudden strong winds and a significant rise in temperature can occur; such storms are common in mountainous areas such as the Canadian Rockies. The equations, including friction, have also been used to look at idealised sea breeze simulations, Cullen [16]. Wu and Blumen [79] have looked at the interaction between the semi-geostrophic equations and the atmospheric boundary layer as has Young [80].

We adopt a typical approach when embarking on looking at a three-dimensional problem on the sphere, and that is to first consider the shallow-water form of the equations. Instead of the full 3-d problem we now have only a 2-d one with the variation in the third direction being represented only by the changing height of the free surface. It has often been found that the major problems in a 3-d model can be found in the equivalent shallow-water problem, such as problems with the co-ordinate poles in numerical methods. Results obtained for the shallow-water problem are often readily generalised to the full 3-d one with little extra effort. We thus start by trying to prove analytically that there exists a unique solution to the semi-geostrophic shallow-water equations on the sphere, chapter 2. We then proceed to develop a robust numerical method for solving the equations, chapter 3, using some of the knowledge and insight we gained in the analytic work. This numerical method is then used to integrate forward in time both idealised and real atmospheric initial data, chapter 4. These simulations are compared with those from the shallow-water equations derived from the usual equations used in atmospheric modelling. The shallow-water simulations provide some insight into the underlying behaviour of the 3-dimensional equations but it is not expected that they will provide any significant new insight into the behaviour of

the real atmosphere. However, the results gained from the simulation of a form of Rossby-Haurwitz wave, see Haurwitz [33] for a description of the true wave, using the semi-geostrophic shallow-water equations on the sphere may do this. The extra understanding we are looking for, it is hoped, will be provided by the full 3-dimensional model. The extension of this work to 3-dimensions on the sphere is outside the scope of this thesis but some comments on this are provided in chapter 5.

We begin by stating the semi-geostrophic equations after first stating the fundamental predictive equations for the atmosphere and describing which approximations are made to obtain the usual equations used in numerical weather prediction, the so called Primitive equations.

1.1 The semi-geostrophic equations.

Our starting point is the frictionless Navier-Stokes equations for a fluid on a rotating sphere of radius a in spherical polar co-ordinates (λ, ϕ, z) where $\lambda \in [0, 2\pi)$, $\phi \in [\pi/2, -\pi/2]$ and z is the distance from the centre of the sphere

$$\frac{D\underline{u}_3}{Dt} = -\alpha\nabla_3 p - 2\Omega\sin\phi\underline{k} \times \underline{u}_3 + \underline{g} + \nu\nabla_3^2\underline{u}_3 \quad (1.1)$$

where α is the specific volume, Ω is the rotation rate of the sphere and the second term on the right-hand-side of the equation is known as the Coriolis term, \underline{k} is the unit vector normal to the surface of the sphere, ν the viscosity coefficient and \underline{g} is the gravitational force including the centrifugal forces,

$$\underline{g} = \underline{g}_a - \Omega\underline{k} \times (\Omega\underline{k} \times \underline{r})$$

where \underline{r} is the particle position measured from the origin at the sphere's centre and \underline{g}_a is the gravitational force. The derivative D/Dt is the usual Lagrangian derivative and

$$\underline{u}_3 = (u, v, w)$$

$$\nabla_3 = \left(\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda}, \frac{1}{a} \frac{\partial}{\partial \phi}, \frac{\partial}{\partial z} \right)$$

with ∇_3^2 the three dimensional Laplacian. We also require the conservation of mass, often referred to as the continuity equation, given by

$$\frac{1}{\alpha} \frac{D\alpha}{Dt} = \nabla_3 \cdot \underline{u}_3 \quad (1.2)$$

We now assume that the atmosphere is a perfect gas, so that

$$p\alpha = RT \quad (1.3)$$

where T is the temperature, and R the gas constant obtained by taking the inverse mass-weighted average of the gas constants for the gases that make up the atmosphere. To complete the equation set we require the first law of Thermodynamics which may be written as

$$C_p \frac{DT}{Dt} - \alpha \frac{Dp}{Dt} = Q \quad (1.4)$$

where Q is a source term and C_p is the specific heat at constant pressure for the gas. We thus have six equations in the six unknowns, u , v , w , α , T and p . In the atmosphere the source term in the thermodynamic equation contains a very large contribution from changes in state of water and hence a predictive equation for the distribution of water in the atmosphere is required,

$$\frac{DL}{Dt} = S \quad (1.5)$$

where L is the total amount of water, in kilograms of water per kilogram of air, and S is a source term. These general equations are simplified by first neglecting the viscous terms. We then make the Boussinesq approximation which allows us to ignore density variations except in the buoyancy terms. This approximation is valid under the conditions of interest provided that the typical scale of any motion is small compared to the density scale height of the atmosphere itself. Thus it says that motions are of shallow depth. This is roughly true for the

troposphere, the bottom 10 kilometres or so of the atmosphere, but is less true for motions above this region. Since most models are concerned with modelling the troposphere this approximation is commonly used. For a full discussion of the Boussinesq approximation see Veronis and Spiegel [74]. Finally the atmosphere is assumed to be in hydrostatic balance

$$0 = -\alpha \frac{\partial p}{\partial z} - \underline{g} \cdot \underline{k} \quad (1.6)$$

Hydrostatic balance is obtained from the equation for the vertical velocity w , when it is assumed that vertical accelerations are negligible. After making these approximations we obtain the Boussinesq Primitive equations which form the basis for almost all NWP models.

The usual approach to understanding this 3-d problem is to first consider the simpler problem obtained by making the shallow-water approximation. The system obtained contains many of the salient features and usually all of the problems encountered in developing numerical methods, such as problems with the coordinate poles. It is also often easy to extend results and methods developed for the shallow-water equations to the full 3-d problem. Since in this thesis we shall be concerned with the shallow-water semi-geostrophic equations on the sphere, and since the extra approximations we shall make to derive the simpler equation sets are equally well illustrated in the shallow-water form as in the full 3-dimensional form, we now make the shallow-water assumption to obtain

$$\frac{D\underline{u}}{Dt} = -g\nabla h - 2\Omega \sin\phi \underline{k} \times \underline{u} \quad (1.7)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\underline{u}) = 0 \quad (1.8)$$

where h is the height of the free surface, $\underline{u} = (u, v)$ and

$$\nabla = \left(\frac{1}{a \cos\phi} \frac{\partial}{\partial \lambda}, \frac{1}{a} \frac{\partial}{\partial \phi} \right)$$

$$\nabla \cdot (h\underline{u}) = \left(\frac{1}{a \cos\phi} \frac{\partial(hu)}{\partial \lambda} + \frac{1}{a \cos\phi} \frac{\partial}{\partial \phi} (\cos\phi hv) \right)$$

and the Lagrangian derivatives are taken along the free surface. Equations (1.7) and (1.8) will be referred to the shallow-water primitive equations, and simulations with them will be used in chapter 4 as a comparison with those obtained from the semi-geostrophic equations. The simple models, which we now derive, are all based on approximations which ‘balance’ the horizontal pressure gradients with some function of the horizontal velocities. For this reason these models are known as *Balanced Models*. The simplest, and most common definition of balance is known as *geostrophic balance* and defines a wind field called the *geostrophic wind*, denoted \underline{u}_g , by

$$\frac{g}{a \cos \phi} \frac{\partial h}{\partial \lambda} = f v_g \quad (1.9)$$

$$\frac{g}{a} \frac{\partial h}{\partial \phi} = -f u_g \quad (1.10)$$

where $f = 2\Omega \sin \phi$ is referred to as the Coriolis parameter. The first set of simple equations considered were the *Quasi-Geostrophic* equations. These can be obtained by forming the vorticity equation and then eliminating the divergence from it using the continuity equation. If we now let the horizontal velocities be geostrophic we obtain the Quasi-Geostrophic equations. This derivation can be found in many textbooks on dynamical meteorology, for example Riegel and Bridger [60] which also contains a summary of the limitations of this equation set. This very simple set of equations was used before modern computers became available to provide understanding about the atmosphere, for example see Charney [11], and was also used as the basis for early numerical weather prediction models. Much work has been done with these equations but they are too simplified to describe many of the important mechanisms involved in atmospheric dynamics.

A less restricted set of equations, still using geostrophic balance, was proposed by Eliassen [27] in 1948 but was largely ignored until the publication of the 1972 paper of Hoskins and Bretherton [38]. Salmon has shown, [64] [65], that the

quasi-geostrophic equations are valid for a particular chosen rest state and that they cease to be so if significant deviation from this state occurs. The equations of Eliassen do not suffer from this problem since they require no prescribed rest state. Since the publication of [38] in 1972 these equations have been extensively used as a tool in understanding mid-latitude dynamics on an f -plane. They have become known as the *Semi-Geostrophic* equations since they are derived by including the geostrophic balance equations in the equation set, but replacing \underline{u} by \underline{u}_g only as the predicted quantity in the momentum equations. This approximation was called the *geostrophic momentum* approximation by Hoskins [36]. The resulting equations when used on the sphere are

$$\frac{D\underline{u}_g}{Dt} = -g\nabla h - f\underline{k} \times \underline{u} \quad (1.11)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\underline{u}) = 0 \quad (1.12)$$

$$g \frac{1}{a \cos \phi} \frac{\partial h}{\partial \lambda} = f v_g \quad (1.13)$$

$$g \frac{1}{a} \frac{\partial h}{\partial \phi} = -f u_g \quad (1.14)$$

where the Lagrangian trajectories are calculated using the full wind field, \underline{u} , rather than the geostrophic wind, \underline{u}_g . These are the equations we wish to study on the sphere and we shall refer to them as the shallow-water spherical semi-geostrophic equations to avoid confusion with other forms of the semi-geostrophic equations on the sphere, see section 1.2 for a discussion of other possible equation sets. Since it has been shown that friction may be included in these equations, Cullen [16], and that this modifies the definition of geostrophic balance, we henceforth denote \underline{u}_g by \underline{u}_0 . This notation is inspired by Hoskins [36] where in his derivation of the semi-geostrophic equations he showed that the full horizontal velocity field \underline{u} can be expanded in terms of the geostrophic velocity \underline{u}_g

$$u = u_g - \frac{\mathcal{D}v_g}{\mathcal{D}t} - \frac{\mathcal{D}^2}{\mathcal{D}t} \left(u_g - \frac{\mathcal{D}v_g}{\mathcal{D}t} \right) + \frac{\mathcal{D}^4}{\mathcal{D}t} \left(u_g - \frac{\mathcal{D}v_g}{\mathcal{D}t} \right) - \dots$$

$$v = v_g + \frac{\mathcal{D}u_g}{\mathcal{D}t} - \frac{\mathcal{D}^2}{\mathcal{D}t} \left(v_g + \frac{\mathcal{D}u_g}{\mathcal{D}t} \right) + \frac{\mathcal{D}^4}{\mathcal{D}t} \left(v_g + \frac{\mathcal{D}u_g}{\mathcal{D}t} \right) - \dots$$

where

$$\frac{\mathcal{D}}{\mathcal{D}t} = \frac{1}{f} \frac{D}{Dt}$$

provided that

$$\left| \frac{Du}{Dt} \right| \ll |f\underline{u}|$$

The geostrophic velocity is thus the leading order approximation to the full velocity and we adopt the notation \underline{u}_0 to represent the approximation of \underline{u} which includes no derivative terms. In a series of papers [21], [22] Cullen and co-workers proved the existence and uniqueness of solutions to the f -plane semi-geostrophic equations for any finite time. The ability to prove such results is an attractive feature of this equation set and naturally poses the question as to whether the results can be extended to the semi-geostrophic equations on the sphere. The proof they obtained was constructive and formed the basis of an ingenious numerical model, the *geometric* model, Cullen and Purser [20], which is guaranteed to converge to the analytic solution of the equations as the spatial and temporal resolution is increased. A finite-difference model was then developed, Cullen [15], which could be compared with the geometric model and allowed more complicated features to be examined. If we can extend the theory to the semi-geostrophic equations on the sphere then we should be in an equally good position to construct an accurate numerical algorithm which also converges to the analytic solution. This model is fundamental to our work as it will be the basis for our attempts to gain understanding, since analytic solutions for problems on the sphere are very scarce. For problems initialised with real atmospheric data, the numerical model is our only way of gaining understanding. At this point we briefly consider some other balanced equation sets.

1.2 Other comparable balanced equation sets.

We have chosen to study the semi-geostrophic equations on the sphere, and we have simply taken the basic definition of the equations and allowed the coriolis parameter f to vary and included spherical geometry. The equations that result do not possess some of the basic properties that the full equations do, for example they do not conserve potential vorticity. To try and retain such properties for varying f both Shutts [68] and Salmon [63] have used a Hamiltonian approach to the geostrophic momentum approximation. As they had different goals in mind when deriving the equations they chose to make different assumptions and thus obtained different equation sets. Salmon's approach is designed to keep exact conservation of energy and potential vorticity whilst allowing order one variations in the Coriolis parameter, f . For the f -plane case his equations are identical to the semi-geostrophic equations. Shutts's approach was to obtain equations valid on the sphere which also conserved energy and potential vorticity. The equations he derives, called the *planetary semi-geostrophic* equations, are identical to the semi-geostrophic equations except for the Lagrangian derivative term in the meridional momentum equation which becomes

$$\sin\phi \frac{D}{Dt} (v_g \sin\phi)$$

These equations allow non-zero height gradients along the equator unlike the semi-geostrophic equations, and thus support a wider class of solutions there. Thus we have three versions of the semi-geostrophic equations for variable f . To avoid confusion we now adopt the following terminology. We refer to the semi-geostrophic equations we shall use as the *spherical semi-geostrophic equations*, those of Salmon as *Salmon's semi-geostrophic equations*, and those of Shutts as the *planetary semi-geostrophic equations*.

It is an open question as to which of the three versions of the semi-geostrophic equations is the *best* one to investigate atmospheric motions on the sphere. It

could be argued that those sets derived from a Hamiltonian approach are in some way better because they preserve some of the properties of the full equations which the spherical semi-geostrophic equations do not. Another advantage of deriving the equations this way is that it introduces the possibility of finding a *symplectic* numerical solution procedure for them. A symplectic numerical method is one which preserves the Hamiltonian structure of the analytic equations in a time integration. Symplectic methods have been shown to produce more accurate results, and in some cases to show that the non-symplectic method has actually produced a completely spurious solution, see for example Miller [51]. These methods are currently being applied to systems of ordinary differential equations but their application to partial differential equations is probably only a matter of time. To help decide which set to look at we could also investigate how well they represent simple solutions, for example linearised wave solutions, when compared to the solutions obtained from the Primitive equations. Shutts [68] showed that for the planetary semi-geostrophic the phase speeds of the gravest modes of travelling wave solutions are greater than those for the corresponding modes of the primitive equations. In chapter 4 we shall consider the phase speeds for the gravest modes of the spherical semi-geostrophic equations and show that they are slower than for the gravest modes of the primitive equations. This demonstrates that the choice of which set to study could lead to quite different answers. As all three equation sets reduce to the same one on an f -plane we expect that the differences between them will be most pronounced in low-latitudes where the variation in f is greatest. However, there might be considerable differences in mid-latitudes as well as the differences in phase speeds for travelling waves suggest. In this thesis we have chosen to study the spherical semi-geostrophic equations, we could equally well have chosen one of the other equation sets. In principle the numerical method that is developed in chapter 3 could be applied to the planetary semi-geostrophic equations, but care would have to be exercised

at the equator where v_g is allowed to be infinite but is multiplied by $\sin\phi$ which is zero. To implement this on a computer will require some very careful thought. Definitions of balance other than geostrophic can be used to define more general equation sets, such as the *linear balance equations* and the *balance equations* described in McWilliams and Gent [45]. Discussion of these equations, along with other balanced equations, can be found in that paper and in others by the same authors [30] and [31] and more recently in Allen et al. [2]. These models are more general than those derived from geostrophic balance but because they describe more atmospheric motions they may be less useful in providing the understanding that we seek. This is because their extra complexity may make it more difficult to isolate the important processes. The logical approach seems to be to extract all the understanding we can from each level of simple model and to gradually increase the models' complexity as our understanding grows. It is thus necessary that we discover what we can from models based on geostrophic balance before embarking on more complex systems.

Chapter 2

Theoretical results

In this chapter we try to prove the existence and uniqueness of solutions to the shallow-water spherical semi-geostrophic equations on the sphere. We begin by reviewing the known results for the 3-dimensional f -plane equations which were proven by Cullen and co-workers [21] [22] and note where the problems lie in extending their theory to the sphere. We then consider a different approach for the shallow-water equations on the sphere and proceed to try to prove the existence and uniqueness of a solution, noting as we go some points which will be relevant in the implementing of a numerical method for these equations. We are unable to prove the required results, but include the work we have done noting what still needs to be done to complete the proofs. Finally we look at how we might extend the solutions beyond the point where the constraints we have identified as being necessary, but possibly not sufficient, to prove existence and uniqueness are violated.

2.1 Known results for the f -plane case

Results for the $f = \text{constant}$ or f -plane case were proved in a series of papers by Cullen and co-workers in the last decade. These proofs were for the 3-dimensional equations in a bounded domain, but can be extended to periodic domains as well. The basis of their approach was a co-ordinate transformation which yields a particularly simple set of equations in the new variables and a Monge-Ampère equation, the solution to which determines the mapping between the co-ordinate systems. We present their method here, without proof, for illustrative purposes and to enable us to draw comparisons later. We shall consider why it is not possible to extend this method to the sphere whilst noting the constraints required to obtain the results, since it is expected that similar constraints will be required to obtain the results on the sphere.

2.1.1 Existence and Uniqueness on a f -plane

Most of this subsection can be found in Cullen and Purser [21] where discussion on the extensions of the theory to more complicated cases, including the equations on the sphere, is also presented. The proofs of the mathematical results they needed can be found in Cullen et al [22].

We begin by stating the 3-dimensional hydrostatic f -plane semi-geostrophic equations. We adopt the notation of Cullen and Purser with the exceptions that we use \underline{u}_0 instead of \underline{u}_g and Φ for ϕ to avoid confusion with the spherical polar co-ordinates (λ, ϕ) used later.

$$\frac{Du_0}{Dt} = -f(v_0 - v) \quad (2.1)$$

$$\frac{Dv_0}{Dt} = -f(u - u_0) \quad (2.2)$$

$$\frac{D\theta}{Dt} = 0 \quad (2.3)$$

$$\frac{D\alpha}{Dt} = 0 \quad (2.4)$$

$$(fv_0, -fu_0, g\theta/\theta_{ref}) = \nabla\Phi \quad (2.5)$$

where $D/Dt = \partial/\partial t + \underline{u} \cdot \nabla$, θ is the thermodynamic variable defined as

$$\theta = T \left(\frac{p_{ref}}{p} \right)^{.286}$$

with θ_{ref} and p_{ref} some reference values, typically $330K$ and $100000pa$ respectively, f is the constant coriolis parameter. The continuity equation is written in terms of the specific volume α . The vertical co-ordinate z is the one introduced by Hoskins and Bretherton [38] which allows the particularly simple form of the continuity equation (2.4). It is defined as follows,

$$z = \left(1 - \left(\frac{p}{p_{ref}} \right)^{\frac{\gamma-1}{\gamma}} \right) \frac{p_{ref}\gamma}{\rho_{ref}g(\gamma-1)}$$

where ρ_{ref} is a reference density value and γ is the ratio of specific heats, see Hoskins and Bretherton for more details. The equations are considered in a closed domain Ω in $\underline{x} = (x, y, z)$ with zero mass flux through the boundary.

A new co-ordinate system is now-defined as follows

$$\underline{X} = (X, Y, Z) = [x + v_0/f, y - u_0/f, g\theta/(f^2\theta_{ref})]$$

and with this definition we can write the equations as

$$\frac{DX}{Dt} = -f(Y - y) \quad (2.6)$$

$$\frac{DY}{Dt} = -f(x - X) \quad (2.7)$$

$$\frac{DZ}{Dt} = 0 \quad (2.8)$$

$$\frac{D\rho}{Dt} = 0 \quad (2.9)$$

where ρ , defined as $\partial(\underline{x})/\partial(\underline{X})$, is referred to as the *inverse potential vorticity* since it can be shown that $\rho = q^{-1}$ where q is the potential vorticity. These new

equations can be re-interpreted as defining the motion of material parcels in the new co-ordinates. Following Cullen and Purser we call $\{\underline{X}\}$ *data space* and $\{\underline{x}\}$ *physical space*. We note that

$$\underline{U}_{\underline{X}} = (f(y - Y), f(X - x), 0)$$

and $\nabla_{\underline{X}} \cdot \underline{U} = 0$ and hence equation (2.9) becomes

$$\left(\frac{\partial}{\partial t} + \underline{U}_{\underline{X}} \cdot \nabla_{\underline{X}} \right) \rho = 0 \quad (2.10)$$

It can be shown that there exists a potential function $R(\underline{X})$ such that $\nabla_{\underline{X}} R = \underline{x}$ and

$$R(\underline{X}) = f^2 \underline{x} \cdot \underline{X} - P(\underline{x})$$

where

$$P(\underline{x}) = \Phi + \frac{1}{2} f^2 (x^2 + y^2)$$

and also that $\nabla_{\underline{x}} P = f^2 \underline{X}$. As pointed out by Chynoweth et al. [13] the co-ordinate transformation is simply a Legendre transform.

The solution procedure is now as follows;

Suppose that at time t we have ρ as a function of \underline{X} subject to the condition that the integral of ρ over all \underline{X} gives the volume of Ω in physical space. We can now solve the following Monge-Ampère equation

$$\det(\partial^2 R / \partial X_i \partial X_j) \equiv \partial(\nabla_{\underline{X}} R) / \partial(\underline{X}) = \rho \quad (2.11)$$

to find R for the given ρ with the boundary condition that $\nabla_{\underline{X}} R = \underline{x}$ is always within Ω for all \underline{X} . After finding R , $\nabla_{\underline{X}} R$ can be calculated and hence the mapping between data space and physical space is established. All the information required to update the solution is now available. It is of note that in this interpretation the full velocities \underline{u} are not used anywhere in the procedure but could be diagnosed if required. They are interpreted as re-arrangements of the parcels and

not simply as advecting velocities. The constraint required to solve the system is that if the potential function P is interpreted as a surface, then it must be convex when viewed from below. This implies that R is also convex. This is equivalent to requiring general two-dimensional parcel stability for the atmospheric state as shown in Shutts and Cullen [69]. The convexity also implies that the potential vorticity q defined as $q = \det(\partial^2 P / \partial x_i \partial x_j)$ is everywhere non-negative. In finite areas where the potential vorticity is zero the fields are still uniquely determined with the exception of the velocity field \underline{u} which is undetermined. This is not a problem since the solution procedure does not require \underline{u} .

The important features of this approach can be summarized as follows;

- i)* The co-ordinate transformation, which allows the system to be written in the simple form given by equations (2.10) and (2.11)
- ii)* The convexity of the function P , which allows the mapping between the two spaces and hence the ability to retrieve the values of all the fields from knowing just ρ .
- iii)* The conservation law for ρ , which allows the integration to proceed for any finite time.
- iv)* The proofs enable discontinuous solutions to be included in the theory since they require only that P and R are continuous; their derivatives need not be.

2.1.2 Application of this Procedure to the Sphere

We consider each of the important features of the approach in turn.

The co-ordinate transformation can again be applied but this now yields terms involving $\partial f / \partial y$ which cannot be absorbed into the D/Dt operator to obtain the same simple form as before. Care must also be taken at the equator where $f = 0$

since the transformation involves terms in $1/f$. Modifying the transformation to have no terms in f in the denominators is easily done and poses no problems in the f -plane case. The fact that the potential vorticity is zero at all points on the equator and $f = 0$ makes the determination of the full velocity field \underline{u} from just knowing ρ not possible there. For the f -plane case it can be shown that if parcels need to be re-arranged to satisfy the constraints, (ii) above, then the values obtained after the re-arrangement do not depend on the path taken. On the sphere these values depend on the path followed and this may not be unique for solutions with jumps. The f -plane proof for $q \geq 0$ uses a ‘global’ re-arrangement property which is not valid on the sphere. If q is bounded away from zero and infinity then \underline{u} is bounded and ‘jumps’ do not occur. So if we desire uniqueness then we will need at least that the potential vorticity is everywhere positive. A further complication is that the conservation law for ρ , and hence also the conservation of potential vorticity, does not hold. The non-conservation of ρ puts a limit on the length of time before the convexity condition is violated which is data dependent. This method would only guarantee existence for an arbitrarily short time, which is not sufficient for practical purposes.

From this we see that we shall need to find a solution procedure which does not require us to write everything in terms of the potential vorticity since we cannot obtain a simple form for this on the sphere. To prove the results for the sphere, stability to small displacements and $q > 0$ away from the equator are likely to be required constraints. We will only be able to prove existence while the constraints hold and this may not be for very long. Therefore it is important that we also consider how we may obtain physically sensible solutions beyond this point. Finally we would like to obtain a solution at the equator which defines, as well as is possible, the values of all the fields there.

2.2 Results for the equations on the sphere

We re-state the shallow-water spherical semi-geostrophic equations (1.11)-(1.14), henceforth SSG, for ease of reference in the rest of this section.

$$\partial h / \partial t + \nabla \cdot (h \underline{u}) = 0 \quad (2.12)$$

$$\partial \underline{u}_0 / \partial t + \underline{u} \cdot \nabla \underline{u}_0 = -g \nabla h - f \underline{k} \times \underline{u} \quad (2.13)$$

$$g \frac{1}{a \cos \phi} \frac{\partial h}{\partial \lambda} - f v_0 = 0 \quad (2.14)$$

$$g \frac{1}{a} \frac{\partial h}{\partial \phi} + f u_0 = 0 \quad (2.15)$$

where \underline{k} is the unit vector in the usual z direction, $f = 2\Omega \sin \phi$, a is the radius of the sphere, g is the gravitational acceleration, h is the height of the free surface, $\underline{u}_0 = (u_0, v_0)$ is the balanced wind field and $\underline{u} = (u, v)$ is the full wind field.

2.2.1 SSG Energy Conservation

We prove that the SSG energy defined as

$$\frac{1}{2} h \underline{u}_0^2 + \frac{1}{2} g h^2$$

is conserved when integrated over the sphere. This proof holds in fact for any closed domain with zero flow across the boundaries. A similar proof for the primitive shallow water equations can be found in Pedlosky [56] where the energy is defined analogously but with the full wind \underline{u} replacing \underline{u}_0 .

Proof:

Taking the dot product of equation (2.13) with $h \underline{u}_0$ gives

$$h \underline{u}_0 \cdot \frac{\partial \underline{u}_0}{\partial t} + h \underline{u} \cdot \frac{1}{2} \nabla (\underline{u}_0^2) + f h \underline{u}_0 \cdot (\underline{k} \times \underline{u}) = -h \underline{u}_0 \cdot g \nabla h \quad (2.16)$$

where we denote $\underline{u}_0 \cdot \underline{u}_0$ by \underline{u}_0^2 . We note the following identity

$$(\underline{A} \cdot \nabla)B = \nabla \cdot (\underline{A}B) - B \nabla \cdot \underline{A}$$

Applying this to the second term in equation (2.16) gives,

$$h \frac{\partial}{\partial t} \left(\frac{1}{2} \underline{u}_0^2 \right) + \nabla \cdot \left(\frac{1}{2} h \underline{u} \underline{u}_0^2 \right) - \frac{1}{2} \underline{u}_0^2 \nabla \cdot (h \underline{u}) + fh \underline{u}_0 \cdot (\underline{k} \times \underline{u}) = -h \underline{u}_0 \cdot g \nabla h \quad (2.17)$$

Equation (2.12) can now be used to simplify the third term in equation (2.17) to

$$\frac{1}{2} \underline{u}_0^2 \frac{\partial h}{\partial t}$$

and this combines with the first term to give

$$\frac{\partial}{\partial t} \left(\frac{1}{2} h \underline{u}_0^2 \right)$$

We now apply equations (2.14) and (2.15) to the fourth term in equation (2.17) to obtain,

$$fh \underline{u}_0 \cdot (\underline{k} \times \underline{u}) = h g \underline{u} \cdot \nabla h$$

Application of equations (2.14) and (2.15) to the term on the right-hand-side of equation (2.17) shows it to be identically zero. Thus we now have,

$$\frac{\partial}{\partial t} \left(\frac{1}{2} h \underline{u}_0^2 \right) + \nabla \cdot \left(\frac{1}{2} h \underline{u} \underline{u}_0^2 \right) = -h g \underline{u} \cdot \nabla h$$

By applying the identity to the right-hand-side of this equation and using equation (2.12) we obtain,

$$\frac{\partial}{\partial t} \left(\frac{1}{2} h \underline{u}_0^2 + \frac{1}{2} g h^2 \right) = -\nabla \cdot \left(\frac{1}{2} h \underline{u} \underline{u}_0^2 \right) - g \nabla \cdot (h^2 \underline{u})$$

Integrating this expression over a closed domain with no flow across the boundaries, and applying the divergence theorem to the right-hand-side gives the desired result.

Q.E.D.

2.2.2 SSG Absolute Vorticity Evolution Equation

The shallow-water form of the SSG potential vorticity denoted q/f , the SSG absolute vorticity, denoted by hq/f , are defined following Hoskins [36] by

$$hq = \left[f + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f \cos \phi} \frac{\partial h}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial h}{\partial \phi} \right] \left[f + \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f} \frac{\partial h}{\partial \phi} \right) \right] - \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f \cos \phi} \frac{\partial h}{\partial \lambda} \right) \right] \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f} \frac{\partial h}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial h}{\partial \lambda} \right]$$

or equivalently,

$$hq = \left[f + \frac{1}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{u_0 \tan \phi}{a} \right] \left[f - \frac{1}{a} \frac{\partial u_0}{\partial \phi} \right] + \left[\frac{1}{a} \frac{\partial v_0}{\partial \phi} \right] \left[\frac{1}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} - \frac{v_0 \tan \phi}{a} \right]$$

Here we show that the SSG equations do not have a Lagrangian conservation law for q/f . The existence of this conservation law was essential in the proof of existence and uniqueness of the f -plane semi-geostrophic equations. To do this we consider the equations on a plane in cartesian space (x, y) with f a function of y . We do this in cartesian co-ordinates rather than spherical polars since it simplifies the proof, as we do not have to include the metric terms, and it is easier to see which terms must cancel in order to obtain conservation of q/f on an f -plane. We wish to emphasize which terms must cancel as this cancellation will be shown to be important in making the semi-Lagrangian version of the numerical scheme work near the poles where f is nearly constant, see Chapter 3, section 3.2.2.

The definition of q in cartesian co-ordinates is

$$hq = \left(f + \frac{\partial}{\partial x} \left(\frac{g \partial h}{f \partial x} \right) \right) \left(f + \frac{\partial}{\partial y} \left(\frac{g \partial h}{f \partial y} \right) \right) - \frac{\partial}{\partial x} \left(\frac{g \partial h}{f \partial y} \right) \frac{\partial}{\partial y} \left(\frac{g \partial h}{f \partial x} \right)$$

or equivalently

$$hq = \left(f + \frac{\partial v_0}{\partial x} \right) \left(f - \frac{\partial u_0}{\partial y} \right) + \left(\frac{\partial u_0}{\partial x} \right) \left(\frac{\partial v_0}{\partial y} \right)$$

and the SSG equations in cartesian co-ordinates are

$$\frac{\partial h}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$\frac{Du_0}{Dt} = -g \frac{\partial h}{\partial x} + fv$$

$$\frac{Dv_0}{Dt} = -g \frac{\partial h}{\partial y} - fu$$

$$g \frac{\partial h}{\partial x} - fv_0 = 0$$

$$g \frac{\partial h}{\partial y} + fu_0 = 0$$

where

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u \frac{\partial}{\partial x} + v \frac{\partial}{\partial y}$$

We now prove that the evolution equation for q on this cartesian plane is,

$$\frac{Dq}{Dt} = \frac{fv\beta}{h} - \frac{u\beta}{h} \frac{\partial u_0}{\partial x} - \frac{v\beta}{h} \frac{\partial u_0}{\partial y} \quad (2.18)$$

where $\beta = \partial f / \partial y$, provided that $h \in C^2$ and $u, v, u_0, v_0 \in C^1$. Where a function is in C^n if all its partial derivatives upto and including order n exist and are continuous.

The evolution equation for q/f is

$$\frac{D(qf^{-1})}{Dt} = f^{-1} \frac{Dq}{Dt} - \beta q f^{-2} v \quad (2.19)$$

which on substituting for Dq/Dt gives a non-zero right-hand-side.

Note: If f is a constant the right-hand-sides of these evolution equations are zero and thus they become conservation laws. Thus we prove the result for the f -plane case which we stated in the previous section.

Proof:

We denote by subscript x the partial differential with respect to x and similarly for y . We define $u_{xy} = (u_x)_y$. Consider,

$$\begin{aligned} \frac{D}{Dt}(v_{0x} - u_{0y}) &= -f(u_x + v_y) - (u_x + v_y)(v_{0x} - u_{0y}) \\ &\quad - u_x u_{0y} + v_y v_{0x} - v_x v_{0y} + u_y u_{0x} - \beta v \end{aligned}$$

We note that we have assumed that $(gh_x)_y = (gh_y)_x$ and thus the terms on the right-hand-side of the equation cancel. We will see the importance of this particular cancellation in chapter 3. Similarly

$$\begin{aligned} \frac{D}{Dt}(-v_{0x}u_{0y} + u_{0x}v_{0y}) &= -(u_x + v_y)(-v_{0x}u_{0y} + u_{0x}v_{0y}) \\ &\quad + fu_x u_{0y} - fv_y v_{0x} - fu_y u_{0x} + fv_x v_{0y} \\ &\quad + gh_{xy}(u_{0y} + v_{0x}) - u_{0x}gh_{yy} - v_{0y}gh_{xx} \\ &\quad - v_{0x}v\beta - u_{0x}u\beta \end{aligned}$$

We note that the terms $u_y v_{0x} u_{0x}$ from $-v_{0x} \frac{D}{Dt}(u_{0y})$ and $u_{0x} \frac{D}{Dt}(v_{0y})$ cancel, and the terms $v_x v_{0y} u_{0y}$ from $-u_{0y} \frac{D}{Dt}(v_{0x})$ and $v_{0y} \frac{D}{Dt}(u_{0x})$ cancel. The terms involving derivatives of h in line 3 of the equation combine to cancel exactly. We now form,

$$\begin{aligned} \frac{D}{Dt}(f^2 + f(v_{0x} - u_{0y}) - v_{0x}u_{0y} + u_{0x}v_{0y}) &= \\ - (u_x + v_y)(f^2 + f(v_{0x} - u_{0y}) - v_{0x}u_{0y} + u_{0x}v_{0y}) & \\ + fv\beta - uu_{0x}\beta - vu_{0y}\beta & \end{aligned}$$

This is simply

$$\frac{D}{Dt}(hq) = q \frac{Dh}{Dt} + \beta(fv - uu_{0x} - vu_{0y})$$

Assuming $h \neq 0$ then dividing through by h and combining the first term on either side of the equals sign we obtain the required result for q .

Q.E.D.

2.2.3 Existence and uniqueness of solutions on the sphere

In this section we would have liked to prove that there exists a unique solution to the SSG equations on the sphere for some finite time under certain conditions. However we have not been able to do this. We present instead an argument which would give us existence and uniqueness if the missing results can be proved. Whether these missing results can be proven is an open question. In chapter 3 we will present some numerical results which suggest that the numerical algorithm is converging to a unique solution. Although this numerical result is not a proof, it suggests that there may exist a unique solution to the SSG equations on the sphere under the conditions we assume in this section. This is because we use the numerical equivalent of these conditions in order to obtain the numerical solution. This gives us some hope that we may eventually be able to prove the analytic existence and uniqueness results.

We introduce the following notation.

Definition 2.1 *We denote the spatial domain by Ω and where we are considering time dependent equations denote the whole domain by $\Omega \times [t_0, t_1]$. When we consider time varying functions we denote the space of continuous functions in space at fixed time t by $C(\Omega)$ and those continuous in time at a fixed point in space by $C[t_0, t_1]$ with functions continuous on the whole domain simply denoted by C . Other function spaces are defined in the same way.*

The Existence and Uniqueness Problem

Suppose we are given $\underline{u}_0(\lambda, \phi, t)$, $\underline{u}(\lambda, \phi, t)$, and $h(\lambda, \phi, t)$ at time $t = t_0$ such that the SSG equations (2.12)-(2.15) are satisfied. We seek to find under what conditions there exists a unique solution to these equations at time $t_1 > t_0$.

The approach we adopt is to first show that the SSG equations can be reduced to one equation in one variable away from $\phi = 0$. We then show that if we insist that

this equation is elliptic that this single equation tends to the natural boundary condition at $\phi = 0$ as we let ϕ tend to zero. This gives us an equation for the single variable at $\phi = 0$ and hence an equation for the whole sphere. We now need to prove that there exists a solution to this equation. This we are not able to do, so we state what needs to be done to prove this result. To obtain the single equation in one variable we have to assume that $\partial h/\partial t$ is continuous in time. To remove this restriction we pose a new problem and then try to prove that there exists a unique solution to this new problem. Again this has not been completed and we show what still needs to be done.

Reduction to a single equation

It is possible to reduce the system of equations (2.12)-(2.15) to one equation for the time evolution of the height field. This can be done by first using equations (2.14) and (2.15) to replace \underline{u}_0 in equation (2.13) provided that $f \neq 0$. This now gives equations for the time evolution of the first derivatives of the height field. The resulting equation, when written out in components, provides a pair of equations which are linear in u and v allowing us to write each of these horizontal velocity components simply in terms of the height field and its first and second derivatives. We can now replace u and v in equation (2.12) provided that q is non-zero, see section 2.2.2 for the definition of q , to obtain the following single equation,

$$\frac{\partial h}{\partial t} + \frac{1}{a \cos \phi} \frac{\partial(hu)}{\partial \lambda} + \frac{1}{a \cos \phi} \frac{\partial(hv \cos \phi)}{\partial \phi} = 0 \quad (2.20)$$

where

$$hu = \frac{1}{q} \left(-\frac{g}{a} \frac{\partial h}{\partial \phi} - \frac{\partial}{\partial t} \left(\frac{g}{af \cos \phi} \frac{\partial h}{\partial \lambda} \right) \right) \left(\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) + f \right)$$

$$+ \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af \cos \phi} \frac{\partial h}{\partial \lambda} \right) \left(-\frac{g}{a \cos \phi} \frac{\partial h}{\partial \lambda} + \frac{\partial}{\partial t} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) \right)$$

and

$$\begin{aligned} hv &= \frac{1}{q} \left(\frac{g}{a \cos \phi} \frac{\partial h}{\partial \lambda} - \frac{\partial}{\partial t} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) \right) \left(f + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{af \cos \phi} \frac{\partial h}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial h}{\partial \phi} \right) \\ &+ \left(\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial h}{\partial \lambda} \right) \left(\frac{g}{a} \frac{\partial h}{\partial \phi} + \frac{\partial}{\partial t} \left(\frac{g}{f a \cos \phi} \frac{\partial h}{\partial \lambda} \right) \right) \end{aligned}$$

We define

$$\begin{aligned} M_x &= f + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{af \cos \phi} \frac{\partial h}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial h}{\partial \phi} \\ M_y &= \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af \cos \phi} \frac{\partial h}{\partial \lambda} \right) \\ N_y &= \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) + f \\ N_x &= \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial h}{\partial \lambda} \\ \dot{h} &= \frac{\partial h}{\partial t} \\ \alpha &= \frac{g}{fq} \end{aligned}$$

With these definitions we observe that

$$hq = M_x N_y - M_y N_x$$

Assuming that all first derivatives of h are continuous then for all $|\phi| > 0$ equation (2.20) is a second order partial differential equation for \dot{h} , with the following form,

$$\begin{aligned} & \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{\alpha N_y}{a \cos \phi} \frac{\partial \dot{h}}{\partial \lambda} \right) - \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{\alpha M_y}{a} \frac{\partial \dot{h}}{\partial \phi} \right) \\ & - \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{\alpha N_x}{a} \frac{\partial \dot{h}}{\partial \lambda} \right) + \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{\alpha M_x \cos \phi}{a} \frac{\partial \dot{h}}{\partial \phi} \right) - \dot{h} \\ & = \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{f \alpha M_y}{a \cos \phi} \frac{\partial h}{\partial \lambda} \right) + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{f \alpha N_y}{a} \frac{\partial h}{\partial \phi} \right) \\ & - \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{f \alpha M_x}{a} \frac{\partial h}{\partial \lambda} \right) - \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{f \alpha N_x \cos \phi}{a} \frac{\partial h}{\partial \phi} \right) \end{aligned} \tag{2.21}$$

which is elliptic if

$$q > \frac{1}{4h}(M_y - N_x)^2 \quad (2.22)$$

Note: It is easily shown that the right-hand-side of this condition is zero if f is constant.

We have reduced the SSG equations to one equation in one unknown everywhere except $\phi = 0$. We now need to complete the reduction to one equation by finding the equation at $\phi = 0$. We note that at $\phi = 0$, $f = 0$ as well and equations (2.14) and (2.15) reduce to

$$\partial h / \partial \lambda = 0$$

$$\partial h / \partial \phi = 0$$

and differentiating with respect to t gives

$$\partial \dot{h} / \partial \lambda = 0$$

$$\partial \dot{h} / \partial \phi = 0$$

assuming that $h \in C^1$. This gives us two first order equations for \dot{h} at $\phi = 0$ instead of one second order equation. We could use these equations at $\phi = 0$ along with equation (2.21) away from $\phi = 0$ and pose the question does there exist a unique solution to these equations on the sphere? Before we contemplate this we investigate the effect that insisting equation (2.21) is elliptic has on h .

Writing q in its component form equation (2.22) is

$$hq = M_x N_y - M_y N_x > \frac{1}{4}(M_y - N_x)^2 \geq 0$$

It would be useful to know what conditions are required on M_x , N_y , M_y , N_x for this to hold. Shutts and Cullen [69] prove that if $h \in C^2$ then a necessary but not sufficient condition for $q > 0$ on an f -plane is that $M_x > 0$ and $N_y > 0$ for $f > 0$ and $M_x < 0$ and $N_y < 0$ for $f < 0$. These conditions are known as

the *inertial stability* conditions. Inertial stability says that parcels are stable to ‘small’ displacements in the stated directions, in this case λ and ϕ , whilst the full criteria $q > 0$ can be considered as requiring inertial stability in all horizontal directions. It is probable that this is also a necessary condition for variable f as well. If we assume that it is then it will also be a necessary condition for equation (2.22) to be satisfied. We now consider how this assumption constrains the solution near $\phi = 0$. If we take just the N_y term then we have,

$$N_y = \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) + f$$

If we insist that N_y is continuous then it must be zero at $\phi = 0$ since it is assumed to be positive for $\phi > 0$ and negative for $\phi < 0$. As we approach $\phi = 0$ then f also approaches zero and we can consider just the remaining term. Since $f = 2\Omega \sin\phi$, $\partial h/\partial \phi = O(\phi)$ as ϕ tends to zero, or this term becomes unbounded. Let us solve

$$\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af} \frac{\partial h}{\partial \phi} \right) = \psi$$

where ψ is a function which goes to zero as ϕ goes to zero. As we are near zero we can approximate $\sin\phi$ by ϕ to an adequate approximation and multiplying through by a^2/g we obtain

$$\frac{\partial}{\partial \phi} \left(\frac{1}{\phi} \frac{\partial h}{\partial \phi} \right) = \psi$$

where we have redefined ψ to include the factor a^2/g . Integrating twice we get

$$h = \int \phi \left[\int \psi d\phi \right] d\phi + c(\lambda)\phi^2 + d(\lambda)$$

Substituting this into the inequality for M_x at $\phi = 0$ tells us that the arbitrary function d can only be a simple constant. Similarly substituting into q at $\phi = 0$ implies that c can also only be a simple constant. Evaluating the integrals for the function ψ is not possible so instead we define

$$\mu = \int \phi \left[\int \psi d\phi \right] d\phi$$

We can write μ in the general form

$$\mu = \phi^r p(\lambda, \phi)$$

for some power r and function p . Substituting this expression into the constraint on N_y to determine the restrictions on r and p gives

$$r(r-2)\phi^{r-3}p + (2r-1)\phi^{r-2}\frac{\partial p}{\partial \phi} + \phi^{r-1}\frac{\partial^2 p}{\partial \phi^2}$$

which must tend to zero as ϕ tends to zero. This is satisfied if $r = 3 + \delta$ for some $\delta > 0$ and $p \in C^2$. Thus the general form of the solution at $\phi = 0$ must be

$$h = d + c\phi^2 + \phi^{3+\delta}p(\lambda, \phi) \quad (\delta > 0) \quad (2.23)$$

where $p \in C^2$.

Substituting this general form for h in equation (2.21) and letting $\phi \rightarrow 0$ gives

$$\begin{aligned} & O(\phi^{-2}) \left[\frac{\partial^2 \dot{h}}{\partial \lambda^2} + \frac{\partial^2 \dot{h}}{\partial \phi^2} \right] + O(\phi^{-3+\delta}) \left[\frac{\partial \dot{h}}{\partial \lambda} \right] \\ & + O(\phi^{-2+\delta}) \left[\frac{\partial^2 \dot{h}}{\partial \lambda \partial \phi} \right] + O(\phi^{-3}) \frac{\partial \dot{h}}{\partial \phi} + O(\phi^0) \dot{h} = O(\phi^1) \end{aligned}$$

If we multiply through by ϕ^3 and look for a solution which is in C^n on the sphere where $n \geq 2$ then this reduces to

$$\frac{\partial \dot{h}}{\partial \phi} = 0$$

at $\phi = 0$. Thus insisting that equation (2.21) is elliptic and taking the limit of it times f^3 as $\phi \rightarrow 0$ gives the same equation at $\phi = 0$ as we obtained by considering the balance equations (2.14) and (2.15). We note that we can multiply equation (2.21) by f^3 away from $\phi = 0$ since f is non-zero and that this does not change the ellipticity condition (2.22).

Proposition 2.2 *There exists a unique solution to equation (2.21) times f^3 on the sphere at time t if*

$$q > \frac{1}{4h}(M_y - N_x)^2 \geq 0 \quad \phi \neq 0$$

$$q = 0 \quad \phi = 0$$

which satisfies

$$\partial \dot{h} / \partial \lambda = 0$$

$$\partial \dot{h} / \partial \phi = 0$$

at $\phi = 0$ with $h \in C^n$ on the sphere with $n \geq 3$. Furthermore the solution $\dot{h} \in C^{n-1}$ on the sphere.

To prove this we need to know when an elliptic equation has a unique solution. The only result I have been able to find in the literature is:

Definition 2.3 *The second-order partial differential equation on a closed domain Ω*

$$\sum_{i,k=1}^n a_{ik}(x_1, \dots, x_n) \frac{\partial^2 p}{\partial x_i \partial x_k} + \sum_{i=1}^n b_i(x_1, \dots, x_n) \frac{\partial p}{\partial x_i} + c(x_1, \dots, x_n)p = r(x_1, \dots, x_n)$$

is elliptic if the quadratic form

$$\sum_{i,k=1}^n a_{ik}(x) \xi_i \xi_k$$

is always definite of the same sign for all $x \in \Omega$, where the ξ_i 's define the coordinate axes.

We shall assume that this quadratic form is *positive* definite and consequently if $A(x)$ is its determinant then $A(x) > 0$ for all $x \in \Omega$. We then have the following Theorem,

Theorem 2.4 *The second-order elliptic partial differential equation on a closed domain Ω*

$$\sum_{i,k=1}^n a_{ik}(x_1, \dots, x_n) \frac{\partial^2 p}{\partial x_i \partial x_k} + \sum_{i=1}^n b_i(x_1, \dots, x_n) \frac{\partial p}{\partial x_i} + c(x_1, \dots, x_n)p = r(x_1, \dots, x_n)$$

has a unique solution $p \in C^2$ for the Dirichlet or Neumann boundary condition on the boundary of the domain, denoted $\delta\Omega$, given by

$$\nu \frac{dp}{dl} + \mu p = \Gamma$$

where l is the normal to the boundary, if the coefficients are all of class C^0 in Ω and r is continuous in Ω and of class C^0 in $\Omega - \delta\Omega$ with $c \leq 0$ and $\mu \geq 0$ but not both identically zero. Further if the coefficients and r are all in $C^{(n-2)}$ then $p \in C^n$.

Proof: See Miranda [52], Section 22, pages 82-85 for the Neumann case and section 36 pages 164-169 for the Dirichlet case.

To apply this theorem to equations on the sphere we need to map the sphere onto a closed domain in cartesian space. If we define

$$x = a\lambda$$

$$y = a\sin\phi$$

then this maps the surface of the sphere to a plane in cartesian space. Applying this transformation to an equation on the surface of the sphere produces the following boundary conditions on the solution p

$$p(0, y) = p(2\pi a, y)$$

$$p(x_1, a) = p(x_2, a) \quad \forall x_1, x_2 \in [0, 2\pi a]$$

$$p(x_1, -a) = p(x_2, -a) \quad \forall x_1, x_2 \in [0, 2\pi a]$$

for which we cannot apply Theorem 2.4. Alternatively we can map the surface of a hemisphere onto a circle in cartesian space by

$$x = a \cos \lambda \cos \phi$$

$$y = a \sin \lambda \cos \phi$$

If we map the northern hemispheric surface onto domain denoted by Ω_N and the southern hemispheric surface onto a domain denoted by Ω_S then the boundary condition is that p on $\delta\Omega_N$ is equal to p on $\delta\Omega_S$. We now show how this mapping onto two hemispheres can be used to try to prove proposition 2.2.

Outline of how Proposition 2.2 might be proved.

Applying the hemispheric mapping to f^3 times equation (2.21) gives an elliptic equation in the open domains Ω_N and Ω_S and on the boundary of each domain the equation becomes

$$\frac{\partial \dot{h}}{\partial l} = 0$$

where l is the normal to the boundary. We could thus apply Theorem 2.4 to both hemispheres separately, *if* the theorem only required the equation to be elliptic on the open domain. However we have not been able to find this result. It then remains to show that the solution on $\delta\Omega_N$ is equal to the solution on $\delta\Omega_S$. If we can show that the solution for each hemisphere is of the form (2.23) then this should not present much of a problem. If the solution is not of this form then we expect that at some later time q will cease to satisfy the ellipticity constraint (2.22). In theorem 2.4 we have that if the coefficients and right hand side are C^n then the solution is in C^{n+2} . To form equation (2.21) we need $h \in C^3$ at least to obtain coefficients in C^0 , and hence the solution for \dot{h} will be at best in C^2 if Theorem 2.4 can be extended to the open domain. Thus we obtain the last line

of the proposition, if $h \in C^n$ then $\dot{h} \in C^{n-1}$.

Finding h knowing \dot{h}

Assuming the existence and uniqueness of a solution to equation (2.21), proposition 2.2, we now have an initial value problem $\dot{h} = F(x, y, t)$, where $F(x, y, t)$ is the solution of equation (2.21). There is a unique solution to this initial value problem for t in some time interval $t_0 \leq t \leq t_1$ provided that F is continuous in this time interval and its first derivatives in space exist and are bounded. (See, for example, [9], chapter 1).

The spatial conditions on F are satisfied by our solution and it thus remains to show that F is continuous in time on this time interval. However we have already assumed this to derive equation (2.21). We would like to remove the assumptions about continuity of h and its derivatives with respect to time. To do this we will consider the following problem,

New Problem

Given equation (2.21) on the sphere, under what conditions does it have a unique solution which is continuous in time for some time interval $[t_0, t_1]$?

We note that if its solution is continuous in time then we can go back from this equation to the SSG equations (2.12)- (2.15), by retracing the steps used to derive this equation, and thus the conditions for the solution of this new problem are exactly those for the solution of the original one. Assuming that q is non-zero and $h \in C^3$ then the only problem in retracing the steps used to derive the single equation comes at $\phi = 0$. Here we can find the balanced velocity field, but the full velocity field is not completely determined at $\phi = 0$. To find \underline{u} at the equator we can solve equation (2.12), but this is only one equation for the two unknowns.

The velocity field \underline{u} can be considered as being made up of two components, a divergent part and a rotational part. If we consider just the divergent part and substitute for \underline{u} using an irrotational form in equation (2.12) then we obtain a Poisson equation. This is easily solved and thus we know the divergent part of the solution. The rotational part is undetermined and hence so is the problem at the equator. Extra information must now be used to obtain the full solution. This situation is not uncommon in many practical problems and the practice of choosing a solution on a physical basis has been shown to work very well, see Seidman [66] for a discussion of this issue. One sensible solution is to choose to set the rotational part of the velocity field to zero at $\phi = 0$.

We would now like to prove the following proposition;

Proposition 2.5 *There exists a unique solution to the New Problem on a domain $\Omega \times [t_b, t_e]$, where Ω is the surface of a sphere radius a , provided that $h \in C^\infty(\Omega)$ at t_b and the following inequality holds at every point in the domain*

$$0 < \frac{1}{4h}(M_y - N_x)^2 < q < \infty$$

where we allow $q = 0$ at $\phi = 0$ and the second partial derivatives of M_y, M_x, N_x, N_y are continuous and bounded with the data inertially stable. Further we then have $h \in C^\infty(\Omega)$ for any $t \in [t_b, t_e]$, and $h \in C^1[t_b, t_e]$ and hence this solution also satisfies the SSG equations.

Outline of how it might be proved.

We consider $t = t_e$ as this implies the result for all the previous values in the interval. We denote the elliptic equation(2.21) by $L(p) = r$ where L is the elliptic operator, p the solution and r the right-hand-side. We note that L, p and r all depend on the time t .

The assumptions we have made guarantee a unique solution to the equation at time $t = t_b$ assuming proposition 2.2. We can form a first estimate to the solution at any other t in the interval by defining,

$$h_1(t) = h(t_b) + p(t_b)t$$

where $p(t_b)$ is the solution of our elliptic equation at time t_b . We note that by construction h is continuous on the time interval and belongs to $C^\infty(\Omega)$, provided that Theorem 2.4 holds for $n = \infty$ which is an open question. Assuming that the q remains bounded, inertially stable and the second partial derivatives of M, N are continuous and bounded, then we could solve the elliptic equation at a later time than t_b and hence refine our estimate. Defining $\Delta t = t_e - t_b$ we can obtain a sequence of approximations defined as follows,

$$h_m(t) = h(t_b) + \sum_{i=0}^{n-1} w_i(t) p_i \frac{\Delta t}{m}$$

where p_i is the solution to the elliptic equation at time $t + i\Delta t/m$ using the values of h_m at this time to calculate the coefficients and right-hand-side, and defining $t_i = t_b + (i - 1)\Delta t/m$ with

$$w_i(t) = \begin{cases} 0 & \text{if } t < t_i \\ 1 & \text{if } t > t_{i+1} \\ (t - t_i)/(t_{i+1} - t_i) & \text{otherwise} \end{cases}$$

It is again noted that provided each of the p_i can be found then this estimate of the solution is also continuous in time and belongs to $C^\infty(\Omega)$ by construction provided that Theorem 2.4 holds for $n = \infty$. We now consider what happens if we let $m \rightarrow \infty$ at time $t = t_e$.

$$h(t_e) = h(t_b) + \int_{t_b}^{t_e} p(\tau) d\tau$$

since $p = \dot{h}$. We noted earlier that for a unique solution to this integral equation we require that p is continuous on the time interval. We can replace the integral

by the limit of the infinite sum,

$$h(t_e) = h(t_b) + \lim_{m \rightarrow \infty} \sum_{i=0}^{m-1} w_i(t_e) p_i \frac{\Delta t}{m}$$

and since we are considering time t_e all the $w_i = 1$. We need to prove that this limit exists and is finite. This we have yet to do. Assuming the boundedness of q , M_x , N_x , M_y , and N_y , should allow us to bound the right-hand-side of the elliptic equation and this may enable us to prove this result. We noted in the outline of how proposition 2.2 might be proved that we required $h \in C^n$ with $n \geq 3$ to form the elliptic equation and that the solution was only in C^{n-1} . Hence if we were to start with $h \in C^3$ we would only be able to solve the elliptic equation since the solution we obtained would then be in C^2 and hence we could not form an elliptic equation from this solution. Since we are taking the limit as $m \rightarrow \infty$ in this procedure we must have $h \in C^\infty$, which is a severe restriction. We have also assumed that q and M_y , M_x , N_x , N_y are continuous and bounded, which is essentially the same as assuming that the derivatives of h are continuous and bounded and thus effectively assuming that h exists. We therefore need to be able to prove these assumptions without assuming h exists. This is possible in the f -plane case since q/f satisfies a Lagrangian conservation law. However as shown in section 2.2.2 this conservation law does not hold for variable f for the SSG equations. It maybe that this proof could be used to prove existence and uniqueness for the Planetary Semi-Geostrophic equations which possess a Lagrangian conservation law for q/f .

2.2.4 Comparison of this Method with other methods for balanced equations

To solve a balanced set of equations the procedure adopted by many authors has been to reduce the set to one predictive equation in one unknown and a set of diagnostic equations to obtain the original variables from this single un-

known. This single predictive equation for the Quasi-Geostrophic equations and the f -plane Semi-Geostrophic equations is simply a Lagrangian conservation law and is thus relatively easily solved. The recovery of the original variables from the new unknown at a later time requires the solution of an equation, a Monge-Ampère equation in the f -plane Semi-Geostrophic case, under some constraint which guarantees a unique solution. See Hoskins et al [39] for a discussion of this idea and how it might be extended to the full equations of fluid motion. We note that in their terminology the recovery of the original variables is referred to as the potential vorticity inversion and the possibility of such recovery the invertability principle. The method used for the SSG equations differs in that the evolution equation is written in terms of one of the original variables and so no recovery is required, unless the direct substitution used to obtain the other variables is considered as such. Effectively the two steps, prediction and recovery, used in other balanced solution procedures have been combined into one with the penalty that the evolution equation has become much more complicated.

Of interest is how large is the time interval for which the solution to the SSG equations would continue to satisfy the constraint (2.22)? This is obviously dependent on the initial data and some estimate of it may be obtainable by considering the SSG vorticity evolution equation and calculating when q would become negative assuming the right hand side of the equation does not change with time. We note that in the f -plane Semi-Geostrophic case the potential vorticity evolution equation is a Lagrangian conservation law and thus we have that the potential vorticity remains bounded indefinitely. In fact, as noted earlier, Cullen et al [22] have shown that it holds for any finite time. We consider in the next section how we may extend the solutions on the sphere beyond the point where the constraint (2.22) is violated by adding additional physical information.

2.3 Solutions beyond the breakdown

Proposition 2.5, which asserts the existence of a unique solution, is not true when q ceases to satisfy the constraint (2.22). As stated already this could happen since the evolution equation for q/f is not a conservation law on the sphere. The question is thus, given any initial data which satisfies the equations can we prove that q is bounded away from zero away from the equator for any finite time? The answer to this is unknown. If we extend the equations to better represent the real atmosphere, by including forcing functions which mimic atmospheric microphysical processes for example, then we expect that we will violate the conditions. We need to define what we mean by a solution at the point where these conditions are violated.

Definition 2.6 *Any solution of the SSG equations belongs to the set of functions, S , where $h \in S$ if $h \in C^3$ and satisfies the constraint (2.22) and is of the form (2.23) about $\phi = 0$.*

We must now find a solution procedure which produces a unique solution within S . In the f -plane case, $f \neq 0$, it has been proved by Cullen et al [21] [22] that given h not in S then there is a unique re-arrangement of $X = fx + v_0$ and $Y = fy - u_0$ such that h then belongs to S . This re-arrangement property does not, unfortunately, extend to the sphere because of the variation of f . In the f -plane case, $f \neq 0$, solutions exist when $q = 0$ whereas on the sphere we have proved existence only for $q > 0$. Assuming that any forcing functions are bounded then it can be shown that if at any time t q becomes zero at some point x then as time continues the solution stays in the solution set by expanding this area of zero q as required. The existence of solutions at $q = 0$ allows uniqueness unlike $q \geq \epsilon > 0$ where the choice of ϵ would determine the solution. For $f^2 \geq \epsilon > 0$ there always exists a solution in S since $h = constant$ would satisfy this. However this solution may not be obtainable via this simple re-arrangement approach

since it does not allow us to change the X and Y values at a material point. This solution may also be unphysical if too large a value for ϵ is chosen. The existence of solutions at $q = 0$ allows the proof of existence of solutions for arbitrary long time via this re-arrangement idea without mixing parcel properties. To obtain a uniqueness result for the sphere a necessary first step appears to be to prove the existence of solutions on the sphere with $q = 0$ away from $f = 0$. For the f -plane case where $f = 0$ then the only stable solution is for h to be a constant on the whole domain. Given a non-constant distribution then the solution is for all the fluid to be mixed to give a homogeneous fluid. For this case a simple re-arrangement of X and Y is not sufficient. On the sphere, effectively the equations have no resolution in the λ -direction at $\phi = 0$ and given a non-constant distribution along $\phi = 0$ then the solution is for all the fluid along $\phi = 0$ to be mixed to give a homogeneous fluid. If we use the geometric model idea of Cullen and Purser [20], where the domain is divided into a finite number of elements with the values of X and Y constant on each element, then it may be that for the sphere a re-arrangement could be proven to exist which preserved the areas of the elements and enforced the extra constraint that there could be only one element at $f = 0$ as this would remove the problem of variations existing along $f = 0$.

Assuming the existence of an analytic unique solution is proved, then in constructing a practical method, primarily for numerical computations, for the sphere these difficulties result in a very ill-conditioned problem with the elliptic equation possibly ceasing to be elliptic at some point. It is thus necessary to stabilize the numerical method in some way. This can be done by restoring the ellipticity constraints when they are violated or if the ill-conditioning is very serious to not allow them to become within some tolerance of being violated. To do this it appears to be sensible to try and include both the approaches already discussed, possibly trying something akin to the re-arrangement procedure applied locally first and

then a global mixing type procedure. It would be desirable to find a unique way of doing this but until the theoretical results exist this is probably not possible. For typical mid-latitude values of f and realistic data the re-arrangements, although ‘global’ in principle, are observed from experiments with the geometric model of Cullen and Purser [20] to be confined to a small area around the points where the semi-geostrophic potential vorticity is negative. As this procedure is known to give a unique answer this seems a viable solution for such mid-latitude values. However near the equator we may need to represent the mixing process and this is typically represented by a diffusion operator which leaves us with a choice of coefficient to make. In the numerical problems attempted in chapter 4 the effect on the solution of varying the coefficient is small, since the areas where mixing is required are almost always near the equator where the solution is almost constant. These results give us some confidence that although the procedure may not give a unique result it is reasonably insensitive to small variations in the choice of parameters. From these experiments we are unable to distinguish between violations of the constraints generated by numerical error and those, if any, generated by the true equations. In the numerical schemes developed in the next chapter we try to minimize the occurrences of computational violations by careful consideration of areas where such errors may be common, for example near the equator. Thus as yet we have no answer to this uniqueness problem, merely numerical procedures which work. This is an area where further work will be required.

Areas of negative potential vorticity correspond to areas where the atmosphere is unstable in some direction. For vertical instability it is often assumed that this instability is removed on a timescale of an hour or so. The SSG equations only represent motions with timescales slower than f^{-1} which is typically about 1 – 2 hours and the faster timescales are collapsed to zero. We know that in

unstable areas the SSG equations are inappropriate to represent such motions and the violation of the constraints is one way of diagnosing where those motions occur. Suppose we consider these equations as acting like a time lapse camera taking pictures of the atmosphere every hour or so. Then what we see as we look at the pictures at one time may be an area of instability, while the next picture shows no instability. By comparing the two we may deduce the net effect of that unstable area, and it is that net effect that we need to represent. The solution method of Cullen, for the f -plane case, allows this to be done exactly since given initial unstable data it constructs the stable minimum energy state which results. However we may wish to add extra physical understanding to this procedure. We could choose to add to our equations an extra set of physical rules, which are only active when instabilities are present and then remove those instabilities leaving us once again in a state where we can continue forward in time. This effect could be included as a forcing term on the right hand side of the equations. As the atmosphere has motions on all scales and is subject to external forcing, from solar radiation for example, such rules already exist for many processes in atmospheric prediction models and are called *Parameterization schemes*. We need to develop such schemes to represent the unstable processes found in the forced equations used here. As this is outside the scope of this work we shall not pursue this matter further. In the next chapter we only discuss how we overcame the problem of instabilities in the numerical model of the unforced equations.

2.4 The Analytic approach as a Numerical Method.

We could use the constructive method given in the outline proof of proposition 2.5, to define a numerical method as follows. We form the discrete analogue

of equation (2.21) and solve the resulting variable coefficient elliptic problem as suggested in the outline proof of proposition 2.2. The solution to this equation can be found via a multigrid solver to be described in section 3.3. Having found this solution we multiply by our chosen timestep to find the new values of h at the next time-level. It now remains to show that these new values also satisfy the constraints of the theorem to allow us to both find the values of the other fields at the new time-level and to continue to advance the solution in time. This procedure has the benefit that the problem is reduced to simply finding $\partial h/\partial t$ and we need only diagnose \underline{u} and \underline{u}_0 if required. The only major problem, therefore, lies in enforcing the constraints should they be violated at the new time-level. Supposing that the constraints are violated, then we could try to avoid this by taking smaller timesteps which would reduce the time truncation error. If the violation is not caused by numerical errors, or if the required timestep would lead to prohibitive computational expense, then we are left having to enforce the constraints directly. We could try the methods to be described in section 3.2.3 which work for the predictor/corrector approach of the next chapter to see if they would restore the constraints. Time has prevented us from pursuing this approach since we began the project by trying to extend the predictor/corrector approach of Cullen [15] and Mawson and Cullen [48].

Chapter 3

Numerical solution procedure

In this chapter we describe a numerical procedure for solving the SSG equations for arbitrarily long time. The approach we used is based on a predictor/corrector method. Here we fix \underline{u} at time t and predict h, \underline{u}_0 at time $t + \Delta t$. The values obtained at $t + \Delta t$ will not satisfy the balance equations and we use them to obtain a correction equation which determines the corrections required to h and \underline{u}_0 in terms of corrections to \underline{u} . This method requires the enforcement of a slightly modified ellipticity constraint from that obtained in the previous chapter. To integrate the equations for an arbitrarily long time we must find a way of enforcing the constraints which guarantee that the corrector equation is elliptic. We describe a procedure for doing this which attempts to minimize the changes to be made to h and \underline{u}_0 to obtain ellipticity. The predictor/corrector method requires both \underline{u} and \underline{u}_0 at the equator to advance the solution in time. As noted in the previous chapter, the rotational part of \underline{u} is not defined at the equator and some care is required in using \underline{u} near the equator. This approach requires the solution of variable coefficient elliptic equations on the sphere and we demonstrate the use of multigrid methods for this purpose.

3.1 The Predictor/Corrector Approach

The basic idea for this method can be found in Cullen [15] where a similar approach was implemented for a 2-dimensional (x, z) f -plane version of the equations. In this section we consider whether we can prove analytically that this approach gives us a unique solution to the SSG equations. We show that for a fixed timestep Δt we can obtain a unique solution via this method if a proposition similar to 2.2 holds, but we are unable to say anything about the convergence as Δt tends to zero. However, we demonstrate, via numerical experiments, that it appears that the method converges to the correct solution for fixed spatial resolution. We begin by re-stating the SSG equations for ease of reference.

$$\frac{\partial h}{\partial t} + \nabla \cdot (h \underline{u}) = 0 \quad (3.1)$$

$$\frac{\partial \underline{u}_0}{\partial t} + \underline{u} \cdot \nabla \underline{u}_0 = -g \nabla h - f \underline{k} \times \underline{u} \quad (3.2)$$

$$\frac{g}{a \cos \phi} \frac{\partial h}{\partial \lambda} - f v_0 = 0 \quad (3.3)$$

$$\frac{g}{a} \frac{\partial h}{\partial \phi} + f u_0 = 0 \quad (3.4)$$

where $f = 2\Omega \sin \phi$, a is the radius of the sphere, g is the gravitational acceleration, h is the height of the free surface, $\underline{u}_0 = (u_0, v_0)$ is the balanced wind field and $\underline{u} = (u, v)$ is the full wind field. We refer to equation (3.1) as the *mass* equation, equation (3.2) as the *momentum* equation and equations (3.3) and (3.4) as the *balance* equations.

For the moment we assume that we are given two distinct problems. Firstly a prediction problem defined by equations (3.1) and (3.2) given \underline{u} . Secondly a re-arrangement or correction problem defined by equations (3.3) and (3.4) given some initial data which does not satisfy them. We seek to find under what conditions unique solutions exist to these two problems. If we can do this we

then consider how these two problems may be combined to prove results for the complete system.

3.1.1 The Advection Problem

We note that equations (3.1) and (3.2) can be written in the following form;

$$M[p] \equiv \frac{\partial}{\partial t} p - \sum_{k=1}^m A_k(x, t) \frac{\partial}{\partial x_k} p - B(x, t)p = r(x, t) \quad (3.5)$$

where $p = (p_1, \dots, p_n)$ is a vector with each p_k a function of $x = (x_1, \dots, x_m)$ and t , $A_k(x, t)$ are Hermitian matrices of degree n . For equation (3.1) $n = 1$ since there is only one variable h , while for equation (3.2), $n = 2$ since there are two components of \underline{u}_0 . $m = 2$ for both equations since we are looking at a problem with two space dimensions. Given u equation (3.1) can be solved independently of equation (3.2) and so we need prove existence and uniqueness for it first and then we may prove existence and uniqueness for equation (3.2) assuming that h is given for the required time interval. It thus suffices to solve the general existence and uniqueness problem for equation (3.5). We assume, without loss of generality, that we start at time $t = 0$ and denote by p_0 the value of p at this time.

Theorem 3.1 *Let A_k be Hermitian and $t \rightarrow A_k(x, t) \in \mathcal{B}^{1+\sigma}$ ($\sigma > 0$) with $t \rightarrow B(x, t) \in \mathcal{B}^1$. Then for an arbitrary $p_0 \in \mathcal{D}_{L^2}^1$ and for an arbitrary $r(t) \in \mathcal{E}_t^0(\mathcal{D}_{L^2}^1)(t \geq 0)$, we have a unique solution of equation (3.5) such that $p(t) \in \mathcal{E}_t^0(\mathcal{D}_{L^2}^1)(t \geq 0)$ and $p \in \mathcal{E}_t^1(L^2)(t \geq 0)$.*

Theorem 3.2 *In Theorem 3.1 if we assume $t \rightarrow A_k(x, t) \in \mathcal{B}^{max(1+\sigma, m)}$ ($0 < \sigma < 1$) and $t \rightarrow B(x, t) \in \mathcal{B}^m$ are continuous, and $p_0 \in \mathcal{D}_{L^2}^m$, $r(t) \in \mathcal{E}_t^0(\mathcal{D}_{L^2}^m)(t \geq 0)$, ($m = 0, 1, 2, \dots$), then there exists a unique solution such that $p(t) \in \mathcal{E}_t^0(\mathcal{D}_{L^2}^m)(t \geq 0)$, where we take the differentiation with respect to t in the sense of the topology of $\mathcal{D}_{L^2}^{m-1}$.*

Proof: The proofs of these theorems can be found in Mizohata [53], where these theorems are referred to as Theorem 6.3 and 6.4 on page 335.

The following notation for certain function spaces is used in the statement of the theorems.

\mathcal{B}^m is the space of functions $f(x)$ where, given $\alpha = (\alpha_1, \dots, \alpha_n)$

$$D^\alpha f(x) = \frac{\partial^{\alpha_1 + \dots + \alpha_n}}{\partial x_1^{\alpha_1} \dots \partial x_n^{\alpha_n}} f(x_1, \dots, x_n) (|\alpha| \leq m)$$

are all bounded and continuous.

$\mathcal{D}(\Omega)$ is the space of C^∞ functions defined on an open set Ω with the support of the function a compact set of Ω .

Let E be a topological vector space. A function belongs to the space $\mathcal{E}_t^m(E)$ ($a \leq t \leq b$) if and only if it is m -times continuously differentiable in $[a, b]$ under the topology of E .

Similarly $\mathcal{E}_{L^p}^m(\Omega)$ is the space of functions defined on an open set Ω , and their derivatives in the sense of distributions belong to L^p including m th order derivatives; the norm of the space is

$$\| f(x) \|_{m, L^p(\Omega)} = \sum_{|\alpha| \leq m} \| D^\alpha f(x) \|_{L^p(\Omega)}$$

(often seen elsewhere as $H_p^m(\Omega)$ or $W_p^m(\Omega)$).

$\mathcal{D}_{L^p}^m(\Omega)$ is a subset of $\mathcal{E}_{L^p}^m(\Omega)$, such that it consists of functions defined as the limits of sequences of functions in $\mathcal{D}(\Omega)$ by the topology of $\mathcal{E}_{L^p}^m(\Omega)$.

With these theorems the existence of solutions to equations (3.1) and (3.2) is proven under the conditions stated.

3.1.2 The Correction Problem

We require that at any given time t_1 equations (3.3) and (3.4) are satisfied. Given any $\underline{u}_0(x, y, t_1)$ and $h(x, y, t_1)$ this will generally not be true and we obtain instead non-zero residuals on the right hand sides.

Question: Do there exist conditions on the given data such that there exists a unique vector $\underline{u}^A(x, y)$ which re-arranges the given data such that equations (3.3) and (3.4) are satisfied ?

Denoting the solution by \underline{u}_0^A, h^A , we construct the desired vector \underline{u}^A as an instantaneous velocity which corrects the given data \underline{u}_0 and h using the SSG advection equations (3.1) and (3.2) giving

$$u_0^A = u_0 - \frac{u^A}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} - \frac{v^A}{a} \frac{\partial u_0}{\partial \phi} + \frac{u^A v_0 \tan \phi}{a} - \frac{g}{a \cos \phi} \frac{\partial (h^A - h)}{\partial \lambda} + f v^A \quad (3.6)$$

$$v_0^A = v_0 - \frac{u^A}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} - \frac{v^A}{a} \frac{\partial v_0}{\partial \phi} - \frac{u^A u_0 \tan \phi}{a} - \frac{g}{a} \frac{\partial (h^A - h)}{\partial \phi} - f u^A \quad (3.7)$$

$$h^A = h - \frac{1}{a \cos \phi} \frac{\partial (h u)}{\partial \lambda} - \frac{1}{a \cos \phi} \frac{\partial (h v \cos \phi)}{\partial \phi} \quad (3.8)$$

The terms on the right-hand-side of equations (3.6) and (3.7) represent, in order of appearance, the value of \underline{u}_0 before the re-arrangement, an advection term which depends on the gradients of \underline{u}_0 , a term representing the effect of changing the height field, and a term due to the sphere's rotation. The term involving the height field correction could be re-written in terms of \underline{u}^A using equation (3.8). Equation (3.8) represents the change to the height field due to the divergence of this instantaneous velocity. We have chosen in equations (3.6) and (3.7) to approximate the advection terms by a simple forward Euler scheme and hence they are written in terms of \underline{u}_0 . We could have used a backward Euler scheme instead, in which case they would have been written in terms of \underline{u}_0^A . In fact, we can choose to represent the advection term by any advection scheme we wish.

The choice of forward Euler is the simplest and gives the simplest form for the resulting correction equation.

If we denote by Δ the difference between the solution and the given data then we can obtain from equations (3.3) and (3.4) the following corrector equations,

$$\frac{g}{a \cos \phi} \frac{\partial \Delta h}{\partial \lambda} - f \Delta v_0 = -R_{ew} \quad (3.9)$$

$$\frac{g}{a} \frac{\partial \Delta h}{\partial \phi} + f \Delta u_0 = -R_{ns} \quad (3.10)$$

where R_{ew} and R_{ns} are the residuals of equations (3.3) and (3.4) for the given data.

We can replace the Δ quantities in these equations by using equations (3.6) and (3.8) to leave two equations in the unknown velocity \underline{u}^A .

$$\frac{g}{a \cos \phi} \frac{\partial D}{\partial \lambda} - f \left(f + \frac{1}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{u_0 \tan \phi}{a} \right) u^A - \frac{f}{a} \frac{\partial v_0}{\partial \phi} v^A + \frac{f g}{a} \frac{\partial D}{\partial \phi} = R_{ew} \quad (3.11)$$

$$\frac{g}{a} \frac{\partial D}{\partial \phi} + f \left(\frac{1}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} - \frac{v_0 \tan \phi}{a} \right) u^A - f \left(f - \frac{1}{a} \frac{\partial u_0}{\partial \phi} \right) v^A - \frac{f g}{a \cos \phi} \frac{\partial D}{\partial \lambda} = R_{ns} \quad (3.12)$$

where we have defined $D = \nabla \cdot (h \underline{u}^A)$. It is now possible to use equation (3.11) to substitute for u^A in equation (3.12) provided that the terms multiplying u^A are non-zero. Similarly we can use equation (3.12) to substitute for v^A in equation (3.11) provided that the terms multiplying v^A are non-zero. Under these assumptions we have

$$\begin{aligned} \left(f - \frac{1}{a} \frac{\partial u_0}{\partial \phi} + \frac{f}{a} \frac{\partial v_0}{\partial \phi} \right) \frac{g}{a \cos \phi} \frac{\partial D}{\partial \lambda} - f q h u^A + \left(f^2 - \frac{f}{a} \frac{\partial u_0}{\partial \phi} - \frac{1}{a} \frac{\partial v_0}{\partial \phi} \right) \frac{g}{a} \frac{\partial D}{\partial \phi} \\ = \left(f - \frac{1}{a} \frac{\partial u_0}{\partial \phi} \right) R_{ew} - \frac{1}{a} \frac{\partial v_0}{\partial \phi} R_{ns} \end{aligned} \quad (3.13)$$

$$g \left(f + \frac{1}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{u_0 \tan \phi}{a} + \frac{f}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} - \frac{f v_0 \tan \phi}{a} \right) \frac{\partial D}{\partial y} - f q h v^A$$

$$\begin{aligned}
& -g \left(f^2 + \frac{f}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{f u_0 \tan \phi}{a} - \frac{1}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} + \frac{v_0 \tan \phi}{a} \right) \frac{\partial D}{\partial x} \\
& = \left(f + \frac{1}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{u_0 \tan \phi}{a} \right) R_{ns} + \frac{1}{a \cos \phi} \frac{\partial u_0}{\partial \lambda} - \frac{v_0 \tan \phi}{a} R_{ew} \quad (3.14)
\end{aligned}$$

If $f \neq 0$ and $q \neq 0$ (see section 2.2.2 for the definition of q), we can multiply both equations (3.13) and (3.14) by $1/fq$. If we take the divergence of the resulting equations, namely

$$\nabla \cdot ((3.13)/fq, (3.14)/fq),$$

we obtain one equation for D of the following form,

$$\begin{aligned}
& \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{A}{a \cos \phi} \frac{\partial D}{\partial \lambda} \right) + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{B_1}{a} \frac{\partial D}{\partial \phi} \right) \\
& + \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{B_2}{a} \frac{\partial D}{\partial \lambda} \right) + \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{C \cos \phi}{a} \frac{\partial D}{\partial \phi} \right) \\
& - D = RHS \quad (3.15)
\end{aligned}$$

where A, B_1, B_2, C are functions of f, h , and first derivatives of $\underline{u_0}$ and RHS is a function of f, h, R_{ew}, R_{ns} and first derivatives of $\underline{u_0}$.

The condition for this equation to be elliptic is,

$$q(1 + f^2) > \frac{1}{4h} \nabla \cdot \underline{u_0} \left(\nabla \cdot \underline{u_0} - 2f \left(f + \frac{1}{a \cos \phi} \frac{\partial v_0}{\partial \lambda} + \frac{u_0 \tan \phi}{a} \right) - 2f \left(f - \frac{1}{a} \frac{\partial u_0}{\partial \phi} \right) \right)$$

Equation (3.15) is similar to equation (2.21) and to complete equation (3.15) on the sphere we have to define it at $\phi = 0$. At $\phi = 0$ equations (3.9) and (3.10) simplify to

$$\frac{g}{a \cos \phi} \frac{\partial \Delta h}{\partial \lambda} = -R_{ew} \quad (3.16)$$

$$\frac{g}{a} \frac{\partial \Delta h}{\partial \phi} = -R_{ns} \quad (3.17)$$

If we insist that the given h is inertially stable and hence satisfies the functional form (2.23) around $\phi = 0$ then it is easy to show that the residuals R_{ew} and R_{ns} are zero at $\phi = 0$. Noting that $\Delta h = -D$ we obtain

$$\frac{\partial D}{\partial \lambda} = 0$$

$$\frac{\partial D}{\partial \phi} = 0$$

$\forall \lambda$ at $\phi = 0$. Multiplying equation (3.15) by $(fq)^2$ and expanding the coefficients proves that equation (3.15) becomes $\partial D / \partial \phi = 0$ as $\phi \rightarrow 0$. If proposition 2.2 can be proven then a similar argument could be used to prove existence and uniqueness for equation (3.15).

This unique D determines, via equations (3.11) and (3.12) a unique \underline{u}^A .

We can now use equations (3.6)-(3.8) to obtain the solution \underline{u}_0^A, h^A which satisfies equations (3.3) and (3.4).

Note: If f is constant and equations (3.3) and (3.4) are satisfied, then the ellipticity condition reduces to $q > 0$, which is a known condition for existence and uniqueness of solutions in this case.

3.1.3 Solving the system by Advection then Correction

We propose a solution method for the system as follows.

Fix $\underline{u}(\lambda, \phi, t) = \underline{u}(\lambda, \phi, t_0)$ for $t \in [t_0, t_1)$ and solve the advection problem, (3.1)-(3.2), to obtain $\underline{u}_0(\lambda, \phi, t_1)$ and $h(\lambda, \phi, t_1)$.

Now solve the correction problem to obtain an instantaneous $\underline{u}^A(\lambda, \phi)$ and hence obtain $\underline{u}_0^A(\lambda, \phi, t_1)$ and $h^A(\lambda, \phi, t_1)$ which satisfy equations (3.3) and (3.4) at time t_1 .

Since both steps have a unique solution we now have obtained a unique solution for \underline{u}_0 and h at time t_1 which satisfies equations (3.1)-(3.4) *but* is dependent on the value of $\Delta t = t_1 - t_0$. To advance our solution to time $t_1 + \Delta t$ we need to obtain \underline{u} at time t_1 .

Define

$$\underline{u}(\lambda, \phi, t) = \underline{u}(\lambda, \phi, t_0) + \underline{u}^A(\lambda, \phi) \left(\frac{t - t_0}{\Delta t} \right)$$

for $t \in [t_0, t_1]$, and hence

$$\underline{u}(\lambda, \phi, t_1) = \underline{u}(\lambda, \phi, t_0) + \underline{u}^A(\lambda, \phi)$$

This definition of $\underline{u}(\lambda, \phi, t_1)$ gives the value which if it had been used at time t_0 would have resulted in no residuals occurring. In effect we are constructing a velocity \underline{u} which is constant for the time period $[t_0, t_1]$. This is better understood by considering u to be defined at the mid-point of this time interval and that it represents an approximation to the actual wind field which gives approximately the same net effect as applying the true wind field for the whole time period. The accuracy of this approximation depends on how fast the true \underline{u} changes with time. For the motions that this equation set represents, the change in \underline{u} is slow and so it should be sufficiently accurate to obtain good solutions. The validity of this statement about accuracy can only be tested by numerical experiments.

Questions :

1. Does the solution obtained by this method converge as Δt tends to zero ?
2. Is the solution to which it converges the true solution of the original problem ?

We consider first the question of convergence. Suppose we are given data at time t_0 . We try to prove that at some fixed time $t_1 > t_0$ the sequence $A_n = h^A(\lambda, \phi, t_1, \Delta t_n)$ converges to a solution A as Δt_n tends to zero, for any fixed choice of λ and ϕ on the sphere, where $h^A(\lambda, \phi, t_1, \Delta t_n)$ is the solution to equations (3.1)-(3.4) at time t_1 obtained by the predictor/corrector method described at the start of this section using a timestep Δt_n , and Δt_n is a strictly monotonic decreasing sequence which tends to zero as n tends to ∞ . We need only show convergence of h^A as this determines the other fields via equations (3.1)-(3.4), provided that

$q \neq 0$ which we have already assumed, and hence they must also converge. Given that we do not know A , to prove convergence it is sufficient to show that the sequence is a Cauchy sequence, i.e.

given $\epsilon > 0$ there exists N such that $\forall m, n > N$,

$$|A_n - A_m| < \epsilon$$

Since $\Delta t_n \rightarrow 0$ as $n \rightarrow \infty$ it is sufficient to show that,

$$|A_n - A_m| < M|\Delta t_n - \Delta t_m|$$

for some bounded constant M .

We cannot obtain an analytic expression for $h^A(\lambda, \phi, t_1, \Delta t_n)$, however, we can construct a numerical method which follows the procedure outlined at the start of this subsection. Thus by first fixing the resolution in space, $\Delta \lambda$, and $\Delta \phi$ or equivalent depending on discretisation method chosen, we can ask the same question of our numerical method (see Section 3.1.4). Showing convergence numerically of the numerical method gives some indication that the analytic method may converge. We are unable at the moment to say more than this.

It then remains to show that this solution is the true solution. A necessary condition for the solution to be a solution of the SSG equations is that R_{ew} and R_{ns} are zero for all time. We therefore need to show that as $\Delta t \rightarrow 0$, then $R_{ew} \rightarrow 0$ and $R_{ns} \rightarrow 0$ at every point in time in the domain. It is sufficient to show that $R \rightarrow 0$ as $\Delta t \rightarrow 0$ where

$$R = \int_{\Omega} \sum_{n=1}^{(t_1-t_0)/\Delta t} (|R_{ew}|_n + |R_{ns}|_n) \Delta t$$

and the domain Ω is taken to be the sphere. R is simply the integral of the timestep-weighted absolute values of the residuals over the time interval. We demonstrate the behaviour of R as the timestep varies in a numerical experiment

in Section 3.1.4. If the advection equations were simply conservation laws of the form

$$\frac{\partial W}{\partial t} + \nabla(\underline{u}W)$$

where W represents any of the variables h , u_0 or v_0 , and the residuals did not contain derivatives, then it would be possible to show that $R \rightarrow 0$ as Δt tends to zero for a simple time-marching algorithm eg: forward Euler. This result is independent of the actual function \underline{u} . As the choice of \underline{u} obviously changes the solution, this choice of \underline{u} is important. Thus to prove $R \rightarrow 0$ as Δt tends to zero it is necessary to bound the right-hand-side of the advection equations with the time truncation error. I do not believe this can be done without knowledge of the function \underline{u} .

3.1.4 Convergence Results derived from Numerical Experiments

A numerical version of the predictor/corrector method was implemented as described in section 3.2. The numerical results given here are for the Heun local-timestepping advection scheme, see section 3.2.2 for details, with two iterations of the corrector step. Similar results are obtained with the other advection schemes.

Convergence to a unique solution

We choose a fixed resolution of 192 by 129 points and take for our initial data real 500hpa fields valid at 2.00 GMT on the 1st February 1991 which we shall call time t_0 . We choose time t_1 to be 3.04 GMT on the same day and obtain the fields valid at this time via the numerical method with a particular choice of the

timestep. We calculate E_{jk} , M_{jk} , E_{jk}^∞ and M_{jk}^∞ defined as follows,

$$E_{jk} = \frac{1}{N} \sum_{i=1}^N |h(\lambda_i, \phi_i, t_1, \Delta t_j) - h(\lambda_i, \phi_i, t_1, \Delta t_k)|$$

$$M_{jk} = E_{jk} / |\Delta t_j - \Delta t_k|$$

$$E_{jk}^\infty = \max_{1 \leq i \leq N} |h(\lambda_i, \phi_i, t_1, \Delta t_j) - h(\lambda_i, \phi_i, t_1, \Delta t_k)|$$

$$M_{jk}^\infty = E_{jk}^\infty / |\Delta t_j - \Delta t_k|$$

where i loops over all N points in the domain, and Δt_j , Δt_k divide $t_1 - t_0$ exactly. E_{jk} is the left-hand-side of our Cauchy inequality, except that it is the L_1 mean over all points, E_{jk}^∞ is the largest value in the domain which typically occurs at the same point for all choices of Δt . M_{jk} and M_{jk}^∞ are the constants on the right-hand-side of the inequalities which we wish to bound to show convergence.

Note: For this time period the special steps taken to enforce the required constants on the data to ensure solutions exist were not required for any of the timesteps.

We choose $\Delta t_1 = 1$ minute. All timesteps in the table are in minutes.

j	Δt_j	E_{1j}	M_{1j}	E_{1j}^∞	M_{1j}^∞
2	2	1.69138×10^{-3}	1.69138×10^{-3}	4.42710×10^{-2}	4.42710×10^{-2}
3	4	5.07192×10^{-3}	1.69063×10^{-3}	0.13299	4.43294×10^{-2}
4	8	1.18311×10^{-2}	1.69015×10^{-3}	0.31020	4.43139×10^{-2}
5	16	2.53300×10^{-2}	1.68867×10^{-3}	0.66529	4.43525×10^{-2}
6	32	5.22773×10^{-2}	1.68644×10^{-3}	1.37714	4.44241×10^{-2}
7	64	1.06399×10^{-1}	1.68887×10^{-3}	2.80488	4.45220×10^{-2}

The same approach was applied to initial data from 20.00 GMT on the 4th June 1992 giving the following results,

j	Δt_j	E_{1j}	M_{1j}	E_{1j}^∞	M_{1j}^∞
2	2	1.69287×10^{-3}	1.69287×10^{-3}	4.43717×10^{-2}	4.43717×10^{-2}
3	4	5.07752×10^{-3}	1.69251×10^{-3}	0.13309	4.43639×10^{-2}
4	8	1.18414×10^{-2}	1.69163×10^{-3}	0.31079	4.43999×10^{-2}
5	16	2.53515×10^{-2}	1.69010×10^{-3}	0.66672	4.44477×10^{-2}
6	32	5.23244×10^{-2}	1.68789×10^{-3}	1.38166	4.45697×10^{-2}
7	64	1.06575×10^{-1}	1.69167×10^{-3}	2.82161	4.47875×10^{-2}

Thus it appears that for these two cases that

$$E_{1j} = M|\Delta t_1 - \Delta t_j|$$

and

$$E_{1j}^\infty = M^\infty|\Delta t_1 - \Delta t_j|$$

where $M = \sup(M_{1j})$ and $M^\infty = \sup(M_{1j}^\infty)$ are bounded, in fact almost constant. Since we cannot run exhaustively all initial data with all timesteps we can never make a definitive statement about numerical convergence. However from these two, randomly chosen, general data cases the results suggest that the method converges to a unique solution for fixed resolution.

It is surprising to note the degree of agreement between the two cases given that the M values derived for each case are data-dependent.

It is trivial to prove that M is data-dependent since for initial data of $h(\lambda, \phi) = \text{constant}$, $\underline{u}_0 = \underline{u} = 0$, the solution remains the same for all time and $M = 0$ for all timesteps regardless of the numerical method.

Convergence of the residuals

For the case of the 4th June 1992 the quantity

$$R_j = \sum_{i=1}^N \sum_{n=1}^{(t_1-t_0)/\Delta t_j} (|R_{ew}|_n + |R_{ns}|_n) \Delta t_j$$

was calculated, where N is the number of grid-points in the domain. The calculations were performed for the same time period and timesteps as Section 3.1.4.

Assuming $R \rightarrow 0$ as $\Delta t \rightarrow 0$ then we can write

$$R = c\Delta t^n$$

for some constants c and $n > 0$. We wish to calculate c and n for this case. For $\Delta t = 1$ we have that $c = R_1$ since

$$R_j = c\Delta t_j^n$$

Using this value for c we calculate n for the other values of j .

j	Δt_j	R_j	n
1	1	2.02×10^{-5}	—
2	2	7.97×10^{-5}	1.98
3	4	3.11×10^{-4}	1.97
4	8	1.19×10^{-3}	1.96
5	16	4.29×10^{-3}	1.93
6	32	1.36×10^{-2}	1.87
7	64	2.89×10^{-2}	1.74

From this it appears that $R \rightarrow 0$ as $\Delta t \rightarrow 0$ and so we believe that the numerical method is converging to the true solution of the SSG equations. As we will prove in section 3.2.2 the Heun scheme as implemented tends to second order accuracy

in time as the timestep tends to zero. The results gained here show that as the timestep decreases R is tending to zero with order Δt^n where $n \rightarrow 2$ as $\Delta t \rightarrow 0$. It thus appears that as the timestep decreases the residuals are tending to zero with the order of the truncation error of the prediction scheme.

3.2 Numerical Implementation of the Predictor/Corrector Method

Notation and Variables

We define the following notation for averaging and centred finite difference operators in cartesian (x, y) and spherical polar (λ, ϕ) co-ordinates where $x = a\lambda\cos\phi$ and $y = a\phi$ for a sphere of radius a . Given an $n \times m$ array of points equally spaced on a domain $[0, X] \times [0, Y]$ then

$$\Delta x = X/(n - 1), \quad \Delta y = Y/(m - 1)$$

and on the sphere given the domain in (λ, ϕ) space as $[0, 2\pi) \times [-\pi/2, \pi/2]$ then

$$\Delta\lambda = 2\pi/n, \quad \Delta\phi = \pi/(m - 1)$$

The operators in cartesian co-ordinates are

$$\delta_x h(x, y) = (h(x + \Delta x/2, y) - h(x - \Delta x/2, y))/\Delta x$$

$$\delta_y h(x, y) = (h(x, y + \Delta y/2) - h(x, y - \Delta y/2))/\Delta y$$

$$\overline{h(x, y)}^x = (h(x + \Delta x/2, y) + h(x - \Delta x/2, y))/2$$

$$\overline{h(x, y)}^y = (h(x, y + \Delta y/2) + h(x, y - \Delta y/2))/2$$

and in spherical polars

$$\delta_\lambda h(\lambda, \phi) = \frac{1}{a\Delta\lambda\cos\phi}(h(\lambda + \Delta\lambda/2, \phi) - h(\lambda - \Delta\lambda/2, \phi))$$

$$\delta_\phi h(\lambda, \phi) = \frac{1}{a\Delta\phi}(h(\lambda, \phi + \Delta\phi/2) - h(\lambda, \phi - \Delta\phi/2))$$

$$\delta_{\phi_c} h(\lambda, \phi) = \frac{1}{a\Delta\phi\cos\phi}(h(\lambda, \phi + \Delta\phi/2)\cos(\phi + \Delta\phi/2) - h(\lambda, \phi - \Delta\phi/2)\cos(\phi - \Delta\phi/2))$$

$$\overline{h(\lambda, \phi)^\lambda} = (h(\lambda + \Delta\lambda/2, \phi) + h(\lambda - \Delta\lambda/2, \phi))/2$$

$$\overline{h(\lambda, \phi)^\phi} = (h(\lambda, \phi + \Delta\phi/2) + h(\lambda, \phi - \Delta\phi/2))/2$$

We also define

$$\underline{U} = (U, V) = (uh, vh)$$

and

$$h(x, y, t + n\Delta t) = h^n(x, y)$$

or simply h^n , with $h^n(\lambda, \phi)$ defined analogously. We note that

$$\overline{\overline{h}^{xy}} = \overline{\overline{h}^{yx}} = \overline{h^{xy}}$$

and that this holds for λ and ϕ also.

3.2.1 Computational Grid Choices

We have chosen to use a finite difference model on a uniformly spaced grid in (λ, ϕ) space. This has the disadvantage of creating problems at the poles of the sphere where we need to hold n polar values for each row, given that we have n points on a row. This is not a serious problem for scalar quantities, such as the height of the free surface h , where all the values are the same but care must be taken with the vector quantities \underline{u}_0 and \underline{u} . In this section we consider the arrangement of the variables on the computational grid and how this affects the accuracy of the discretised equations, as well as identifying any spurious solutions generated by a particular grid choice. We also consider how to define any vector

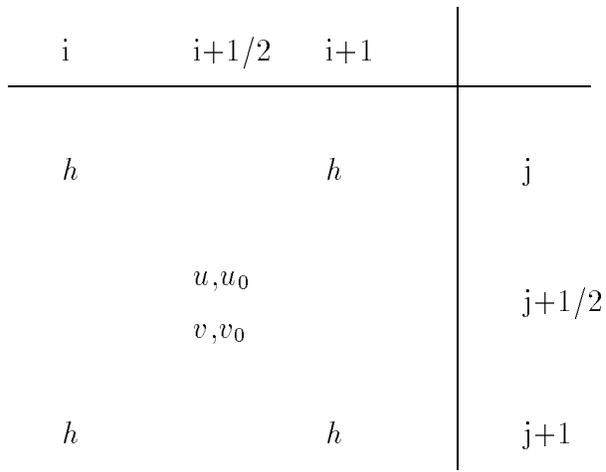


Figure 3.1: SSG model B-grid variable staggering.

quantities at the poles should the grid staggering require their storage there. In computational fluid dynamics and numerical weather prediction much time has already been spent on investigating various grid staggerings, see for example Arakawa [4], and here we shall consider just two. Using the nomenclature of Arakawa and Lamb [5], which classifies the grids depending on the positions of u and v relative to the height of the free surface h , we refer to the two grids as the *B-grid*, fig 3.1, and the *C-grid*, fig 3.2. If we use the semi-Lagrangian advection scheme described in section 3.2.2 then the advection operator in the momentum equation does not influence the grid choice. This leaves us to consider the forcing functions on the right hand side of the momentum equations, the mass equation, and the ability to represent and remove residuals in the corrector step.

The mass equation is

$$h^{n+1} - h^n = -\Delta t (\overline{\delta_\lambda U^\phi} + \overline{\delta_{\phi_c} V^\lambda})$$

on the B-grid and

$$h^{n+1} - h^n = -\Delta t (\delta_\lambda U + \delta_{\phi_c} V)$$

i	i+1/2	i+1	
h	u, v_0	h	j
v, u_0		v, u_0	$j+1/2$
h	u, v_0	h	$j+1$

Figure 3.2: SSG model C-grid variable staggering.

on the C-grid. The B-grid involves averaging operators and so we expect some extra smoothing to be introduced which is not present in the C-grid version. The forcing functions on the right hand side of the momentum equations are $f(v - v_0)$ and $f(u_0 - u)$ for the B-grid and $fv - \overline{(fv_0)}^{\lambda\phi}$ and $\overline{(fu_0)}^{\lambda\phi} - fu$ for the C-grid, where we have used the balance equations (3.3) and (3.4) to substitute for the spatial derivatives of h in the forcing functions. For geostrophically balanced flow $\underline{u} = \underline{u_0}$ and the forcing function should be exactly zero. This is satisfied exactly by the B-grid but not by the C-grid and thus the B-grid will give a more accurate representation of the true forcing for such flows.

For the calculation and removal of the residuals we first consider the C-grid. The residuals are given uniquely without any averaging by

$$R_{ew} = g\delta_\lambda h - fv_0$$

$$R_{ns} = g\delta_\phi h + fu_0$$

To obtain the correction equation we form the right hand side of the equation

$$R = \delta_\lambda R_{ew} + \delta_\phi R_{ns}$$

which is at the “ h ” points on the grid as is the divergence D . Using the centred finite difference operators to discretise the second order partial differential equation yields a 5 point stencil for D which is the minimum possible. We can solve this for D and calculate the implied \underline{U} exactly and thus when we form

$$\delta_\lambda U + \delta_{\phi c} V$$

we obtain the calculated D exactly. In the theory we could remove all the residual in one application but here there is some averaging in the coefficients of equation (3.15) and in certain terms in equation (3.6) which forms the implied corrections to \underline{u}_0 . This averaging produces only very small errors since almost all terms in calculating h and \underline{u}_0 are evaluated exactly using the exact values of D and \underline{U} . However this averaging forces us to perform one or two iterations of the procedure to remove the residuals completely.

For the B-grid we have no unique way to define the residual which involves no averaging. In obtaining the exact forcing for geostrophically balanced flows we have assumed that

$$\overline{\delta_\lambda h^\phi} = f v_0$$

and

$$\overline{\delta_\phi h^\lambda} = -f u_0$$

To be consistent we should use these definitions to define the residuals.

$$R_{ew} = g \overline{\delta_\lambda h^\phi} - f v_0$$

$$R_{ns} = g \overline{\delta_\phi h^\lambda} + f u_0$$

This has the disadvantage that the residuals will not see a chequerboard pattern in the height field. It is well known that schemes on the B-grid suffer from such spurious computational modes and the inability of the correction step to see them, and hence to remove them, is a problem. The mode can be suppressed by

projecting the height field onto the unstable mode either locally or globally and then removing a percentage of the mode found from the data, see Margolin et al. [46]. The choice of percentage used is usually found by experiment. For the local procedure we start at one corner of the grid and sweep through the points removing the mode as we go. In this procedure a small percentage is removed, for example 2%, and a number of sweeps are applied. In the global procedure, as all points are updated simultaneously, a larger percentage, for example 10%, is used and only one application is required. This procedure, as well as suppressing the checkerboard mode, can add some additional smoothing to the solution. To define a 5 point stencil for D we need to define D at the u, v points on the grid, not the h point where it is defined for the mass equation. We also need to take centred differences of the residuals over two grid-lengths, not one, since the residuals are now also held at the \underline{U} points and not half a grid-length away in the appropriate direction as they are for the C-grid.

$$R = \delta_{2\lambda} R_{ew} + \delta_{2\phi} R_{ns}$$

We thus end up with a much larger stencil for the right hand side than we did for the C-grid and this leads to a smoother, less accurate solution of the elliptic equation on the first iteration than obtained with the C-grid. There is also some averaging in the calculation of the coefficients of the partial differential equation as we found with the C-grid. Having found the solution D we now find the corresponding \underline{U} and to do this we must calculate the derivatives of D found in equations (3.11) and (3.12) centred over two grid-lengths. The major drawback of the B-grid is that the D used in the calculation of the height correction is an averaged version of the one used in the balanced velocity \underline{u}_0 update.

$$h^A = h - \delta_{2\lambda} D - \delta_{2\phi_c} D$$

We find that the height field correction is much smoothed because of this, typically 25 – 50% of the true correction, whilst the balanced velocity correction is almost

exact. This procedure, as for the C-grid, needs iterating and because of the smoothing of the height field update, it typically takes many iterations to converge to the same accuracy as the C-grid solution obtains without iteration. To derive the same accuracy as the C-grid procedure involves at least an order of magnitude more iterations and this is computationally very expensive. This inability to remove the residuals efficiently along with the computational checkerboard mode has convinced us that we should not solve the equations on the B-grid.

So far we have not considered the effect of the grid choice on which variables are held at the poles and on the equator. We consider just the C-grid since we have discarded the B-grid. In the last chapter we required $\partial h/\partial t$ on the equator to solve the elliptic equation (2.21). We thus choose to put h , and hence D , on the equator enabling us to solve the elliptic equation which arises in the correction set via a similar approach to that used for equation (2.21). This choice implies that we hold U at the equator as well. We determine the divergent part of U at the equator, see section 3.2.4 for details, whilst the rotational part of U is set to zero. The inertial stability constraint requires that $v_0 = 0$ on the equator and hence v_0 on the equator is zero for all time. We choose to put a height point at each of the poles as well since this allows an easy definition of the divergence at the pole as the sum of the polewards mass fluxes V . The height value at the pole represents the value for the polar cap which extends half a grid-length equatorwards of the pole in all directions. We also need to define v_0 at the pole and instead of solving the momentum equation for v_0 at the pole we choose to derive its value from surrounding values. This is done by first calculating a mean local cartesian \underline{u}_0 at the pole using the values of u_0 half a grid-length equatorwards and v_0 one grid-length equatorwards as follows;

$$u_{0c}^{np} = \frac{1}{n} \sum_{i=1}^n \left[\overline{u_0((i-1/2)\Delta\lambda, \pi/2 - \Delta\phi)}^\lambda \cos(i\Delta\lambda) - v_0(i\Delta\lambda, \pi/2 - \Delta\phi/2) \sin(i\Delta\lambda) \right]$$

$$v_{0c}^{np} = \frac{1}{n} \sum_{i=1}^n \left[\overline{u_0((i-1/2)\Delta\lambda, \pi/2 - \Delta\phi)}^\lambda \sin(i\Delta\lambda) + v_0(i\Delta\lambda, \pi/2 - \Delta\phi/2) \cos(i\Delta\lambda) \right]$$

$$u_{0c}^{sp} = \frac{1}{n} \sum_{i=1}^n \left[\overline{u_0((i-1/2)\Delta\lambda, -\pi/2 + \Delta\phi)}^\lambda \cos(i\Delta\lambda) + v_0(i\Delta\lambda, -\pi/2 + \Delta\phi/2) \sin(i\Delta\lambda) \right]$$

$$v_{0c}^{sp} = \frac{1}{n} \sum_{i=1}^n \left[-\overline{u_0((i-1/2)\Delta\lambda, -\pi/2 + \Delta\phi)}^\lambda \sin(i\Delta\lambda) + v_0(i\Delta\lambda, -\pi/2 + \Delta\phi/2) \cos(i\Delta\lambda) \right]$$

where $\underline{u_0^{np}}$ and $\underline{u_0^{sp}}$ denote the mean cartesian balanced wind at the north and south poles respectively and n is the number of points on a row. This mean local cartesian value is then used to define v_0 at each point around the pole as follows.

$$v_0^{np}(\lambda) = v_{0c}^{np} \cos \lambda - u_{0c}^{np} \sin \lambda$$

$$v_0^{sp}(\lambda) = v_{0c}^{sp} \cos \lambda + u_{0c}^{sp} \sin \lambda$$

Defining v_0 at the poles this way means that U at the pole is required only for the advection of u_0 in the Heun advection step and the calculation of the departure points in the semi-Lagrangian advection step. We discuss its definition for those purposes in the relevant sections. We need also to consider what we mean by $\cos \phi$ at the pole. Analytically this value is zero, which would cause a problem in taking $\delta_{\phi_c} V$ at the pole which is required to calculate the divergence there. This quantity represents the change in mass of the grid box surrounding ϕ by the mass-flux into it from the grid boxes to the north and south. The cosine values in this derivative represent the area of the grid-boxes which are latitude dependent. The cosine value for any grid box can be found by calculating the area which is

$$\int_{\phi-\Delta\phi/2}^{\phi+\Delta\phi/2} \Delta\lambda \cos \mu d\mu = 2\Delta\lambda \cos \phi \sin(\Delta\phi/2)$$

For small values of $\Delta\phi$ we have $\sin(\Delta\phi/2) = \Delta\phi/2$ to a good approximation, $O(\Delta\phi/2)^3$, and hence the area is approximately

$$\Delta\phi\Delta\lambda\cos\phi,$$

and so the cosine which represents the area is almost exactly the analytic value of the cosine at the mid-point of the grid box. If we evaluate this integral at the pole we find that the cosine value which represents the area is $\cos(\pm\pi/2 \mp \Delta\phi/4)$ to the same degree of approximation. This is the value that we need to use at the poles and not the analytic value of zero.

3.2.2 Prediction Schemes

We consider two different numerical schemes for the prediction of h and \underline{u}_0 at time $t + \Delta t$ given h, \underline{u}_0 and \underline{u} at time t . The first is an explicit *Eulerian* advection scheme which has been used for many years and is currently the basis of the U.K. Meteorological Office operational numerical weather prediction model. The second is a *semi-Lagrangian* scheme which has been successfully applied to the primitive equations by many authors, for example McDonald and Bates [44], and shown to have some advantages over standard Eulerian schemes. We wish to see if there is any benefit in applying the semi-Lagrangian approach to the SSG equations. In this section we describe both schemes, emphasising the problems caused by implementation on the sphere, and give accuracy and stability results where possible.

A. The Heun Scheme

The *Heun* advection scheme is a two-time level explicit Eulerian method and is described in Mesinger and Arakawa [50] and Mesinger [49]. To illustrate the

scheme and investigate its temporal accuracy we consider just the simple advection equation,

$$\frac{\partial h}{\partial t} = -c \frac{\partial h}{\partial x} \quad (3.18)$$

The first step of the Heun advection scheme is to form a first approximation of h^{n+1} , denoted $h^\#$, as follows

$$h^\# = h^n - \Delta t c^n \frac{\partial h^n}{\partial x}$$

A second step is now performed using $h^\#$ as the starting value to obtain $h^{2\#}$ by

$$h^{2\#} = h^\# - \Delta t c^n \frac{\partial h^\#}{\partial x}$$

The full update is now given by

$$h^{n+1} = h^n + \left[\frac{1}{2} (h^\# - h^n) + (h^{2\#} - h^\#) \right]$$

which simplifies to

$$h^{n+1} = h^n - c^n \Delta t \frac{\partial h^n}{\partial x} + \frac{1}{2} (c^n)^2 \Delta t^2 \frac{\partial^2 h^n}{\partial x^2} + \frac{1}{2} c^n \Delta t^2 \frac{\partial c^n}{\partial x} \frac{\partial h^n}{\partial x} \quad (3.19)$$

where the last term is zero if c^n has no spatial dependence.

Temporal Accuracy

To investigate the temporal accuracy of the scheme consider the Taylor series expansion of $h(t)$, truncated at the third derivative, which is

$$h^{n+1} = h^n + \Delta t \frac{\partial h^n}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 h^n}{\partial t^2} + O(\Delta t^3)$$

substituting for $\partial h / \partial t$ from the advection equation (3.18) to get

$$h^{n+1} = h^n - c^n \Delta t \frac{\partial h^n}{\partial x} + \frac{1}{2} (c^n)^2 \Delta t^2 \frac{\partial^2 h^n}{\partial x^2} + \frac{1}{2} c^n \Delta t^2 \frac{\partial c^n}{\partial x} \frac{\partial h^n}{\partial x} - \frac{1}{2} c^n \Delta t^2 \frac{\partial c^n}{\partial t} \frac{\partial h^n}{\partial x} + O(\Delta t^3)$$

By comparing this with equation (3.19) we see that if c is independent of x and t then the Heun scheme is second order accurate in time. If c is not independent

of x and t , as is the case for the general advection equations we are using in the SSG model, then the scheme is still first order accurate and tends to second order accuracy as c tends to a constant.

Spatial Accuracy

If we use centred finite differences then we obtain second order spatial accuracy. If higher order accuracy is required it can be achieved simply by broadening the stencil to include more points and using standard higher order centred approximations to the derivatives. In this section we limit ourselves to second order accuracy for simplicity. See Cullen and Davies [18] for an example of a fourth accurate scheme.

Von Neumann Stability

We consider the Heun scheme applied to our simple advection problem (3.18) and we define a wave solution $h^n = \Lambda^n h_0 e^{ikx}$ where Λ is an amplification factor and h_0 is some initial value of h . For stability we require that $|\Lambda| \leq 1$. Substituting the wave solution into the Heun scheme gives

$$\Lambda = 1 - ikc\Delta t - k^2 c^2 \Delta t^2 / 2$$

and hence

$$|\Lambda|^2 = 1 + (kc\Delta t)^4 / 4$$

which is unconditionally unstable except for the trivial case where $c = 0$. Cullen and Davies [18] show that the modified Heun step

$$h^\# = h^n - \Delta t \left(c \frac{\partial h^n}{\partial x} - \frac{\Delta t}{3} \frac{\partial}{\partial x} \left(c^2 \frac{\partial h^n}{\partial x} \right) \right)$$

which includes a diffusion term, has amplification factor

$$\Lambda = 1 - ikc\Delta t - (kc\Delta t)^2 / 2 + i(kc\Delta t)^3 / 6$$

with

$$|\Lambda|^2 = 1 - (kc\Delta t)^4/12 + (kc\Delta t)^6/36$$

which is stable for $kc\Delta t < \sqrt{3}$; see Marshall [47] for examples of these schemes applied to simple problems. In practice if $kc\Delta t < 1$ then the instability in the original scheme is very weak and in more complicated equations where other damping operators are found it is often unnecessary to include this explicit diffusion term. In the implementation in the SSG model we use the scheme in its original form with $kc\Delta t < 1$ and rely on the correction step to overcome the weak instability still present.

Application to the SSG equations

a) Cartesian geometry.

The discretised mass equation (3.1) is

$$h^{n+1} - h^n = -\Delta t(\delta_x U^n + \delta_y V^n) \quad (3.20)$$

which is a simple one step Euler scheme with first order accuracy in time, second order accuracy in space. Alternatively the mass equation could be discretised as

$$h^{n+1} - h^n = -\Delta t(\overline{u\delta_x h^x} + \overline{v\delta_y h^y}) - \Delta t(h\delta_x u + h\delta_y v) \quad (3.21)$$

where the first term on the right hand side could be replaced by a Heun advection step. This has the disadvantage of creating errors in the calculation of the divergence D due to an inconsistency between predictor and corrector steps, because in the correction step the divergence is calculated via the discretisation (3.20) not the alternative one (3.21). The use of the alternative discretisation also leads to large errors at the equator since we now need to use \underline{U} when only the divergent part of \underline{U} is well defined. We thus use the original discretisation for the mass

equation.

The discretised momentum equations consist of an advective part and two forcing terms on the right hand side

$$u_0^{n+1} = H(u_0^n) - \Delta t(g\delta_x \bar{h}^y - fV/\bar{h}^y)$$

$$v_0^{n+1} = H(v_0^n) - \Delta t(g\delta_y \bar{h}^x + fU/\bar{h}^x)$$

where $H(u_0^n)$ represents the Heun advection of u_0^n given at each step by

$$u_0^\# = u_0^n - \left(\overline{\left(\frac{U}{\bar{h}^x} \right)^y} \delta_x u_0^n \right) - \frac{V}{\bar{h}^y} \delta_{2y} u_0^n$$

and $H(v_0^n)$ represents the Heun advection of v_0^n given at each step by

$$v_0^\# = v_0^n - \frac{U}{\bar{h}^x} \delta_{2x} v_0^n - \left(\overline{\left(\frac{V}{\bar{h}^y} \right)^x} \delta_y v_0^n \right)$$

These discretisations are second order accurate in space. Temporal accuracy is only first order since, although the Heun scheme is at least first order accurate and tends to second order accurate as \underline{U} tends to a constant, the forcing terms are only first order accurate in time. If we perform a Von Neumann stability analysis for the Heun part of the discretisation only, assuming a wave solution of the form $h^n = \Lambda^n h_0 e^{i(kx+ly)}$, we find that

$$|\Lambda|^2 = 1 + \alpha^4/4$$

as before except that here

$$\alpha = \frac{u\Delta t}{\Delta x} \sin(k\Delta x) + \frac{v\Delta t}{\Delta y} \sin(l\Delta y)$$

b) Spherical Polar geometry.

The discretised equations now have to include the effects of the co-ordinate transformation and become

$$h^{n+1} - h^n = -\Delta t(\delta_\lambda U^n + \delta_{\phi_c} V^n)$$

$$u_0^{n+1} = H(u_0^n) - \Delta t(g\delta_\lambda \bar{h}^\phi - fV/\bar{h}^\phi)$$

$$v_0^{n+1} = H(v_0^n) - \Delta t(g\delta_\phi \bar{h}^\lambda + fU/\bar{h}^\lambda)$$

where $H(u_0^n)$ represents the Heun advection of u_0^n given at each step by

$$u_0^\# = u_0^n - \overline{\left(\left(\frac{U}{\bar{h}^\lambda}\right)^\phi \delta_\lambda u_0^n\right)^\lambda} - \frac{V}{\bar{h}^\phi} \delta_{2\phi} u_0^n + \overline{\left(\frac{U}{\bar{h}^\lambda} v_0^n\right)^\lambda} \frac{\tan\phi}{a}$$

and $H(v_0^n)$ represents the Heun advection of v_0^n given at each step by

$$v_0^\# = v_0^n - \frac{U}{\bar{h}^\lambda} \delta_{2\lambda} v_0^n - \overline{\left(\left(\frac{V}{\bar{h}^\phi}\right)^\lambda \delta_\phi v_0^n\right)^\phi} - \frac{U}{\bar{h}^\lambda} u_0^{n\lambda\phi} \frac{\tan\phi}{a}$$

The choice of averaging of the trigonometric terms involving $\tan\phi$ which arise from the co-ordinate transformation is arbitrary and the one shown here is certainly not the only choice. It is, perhaps, the most natural choice and experiments with other averagings did not demonstrate any significant sensitivity to the choice. The accuracy properties of the discretisation are not affected by the co-ordinate transformation, which just leaves the question of stability. The Von Neumann stability analysis performed previously requires the ratio of Δx to Δy to remain fixed which is not true in spherical geometry. Therefore we cannot perform the analysis and hence we cannot investigate the choice of the averaging of the trigonometric terms on stability. Again experiments have shown little sensitivity to the choice. Assuming that the linear analysis would have yielded the same amplification factor, which numerical experiments demonstrate to be approximately true, then problems will arise as we approach the poles, since $u\Delta t/a\Delta\lambda\cos\phi$ will typically be an order of magnitude bigger than the equatorial values. If, as experiments suggest, we require that this quantity remains less than 1 everywhere for a stable solution then we obtain a severe restriction on the timestep we can choose. We propose two different approaches to relax this constraint on the timestep.

i) Local Time-stepping.

The idea is to choose a timestep Δt which gives a stable prediction scheme at low latitudes and then perform multiple applications of the advection step with a correspondingly shorter, and hence stable, timestep at higher latitudes. Each of these advection steps uses the values of u_0 from the end of the previous step but keeps the advecting velocity \underline{U} fixed at the old time-level value. We only need multiple applications of the Heun scheme in the λ -direction since the stability restriction is not violated in the ϕ -direction for typical atmospheric flows. We calculate the integer number of applications of the Heun scheme, s_ϕ , required on any row ϕ to ensure that $u\Delta t_\phi/a\Delta\lambda\cos\phi < 1$, where Δt_ϕ is the local timestep defined below, by

$$s_\phi \geq 1 + \max_{\lambda, i=\phi-\Delta\phi, \phi+\Delta\phi} |u(\lambda, i)| \frac{\Delta t}{a\Delta\lambda\cos\phi}$$

where we could have used only the maximum value of u on the row ϕ . However experiments showed that using the average over that row and the ones to the north and south produces a more stable solution. This form also allows this value of s_ϕ to be used as the number of applications of the Heun scheme in the prediction of u_0 on the row polewards of this v_0 row. The corresponding *local timestep* for each row is then given by

$$\Delta t_\phi = \Delta t/s_\phi$$

We note that because of the trigonometric term in the momentum equations we should predict both u_0 and v_0 simultaneously so that we always have both variables at all the required intermediate time-levels. Unfortunately, because of the averaging required on the C-grid, this implies that we must perform the same number of applications of the λ -directional Heun scheme on every row to achieve this goal. The scheme was implemented by finding the maximum number of applications required by any row and using this value on all rows. Attempts to

remove this restriction were not made since it was seen from experiments that multiple applications of the Heun scheme in the λ -direction only were less accurate than multiple applications of the whole Heun step. So for momentum advection we implemented multiple applications of the Heun advection using the largest timestep which is stable at all points on the sphere and which divides the desired timestep exactly. This, although not the cheapest option, is cheaper than applying this small timestep to the whole predictor/corrector method, however it is less attractive than it first appeared.

ii) Fourier Damping.

This approach is based on an idea which has been in common use since the beginning of numerical weather prediction, see for example Arakawa [3]. The idea is to choose the timestep based on stability considerations at low latitudes as before but now we check every row on the sphere to find if it has any unstable waves and any we find are then multiplied by a damping coefficient. Effectively we are searching for the waves for which the amplification factor Λ satisfies $|\Lambda| > 1$ and we then wish to multiply those waves by a factor d such that $d|\Lambda| < 1$. These modified waves are now stable and we regain a stable scheme. To do this it is necessary to transform our grid-point data into a wave representation via a Fourier transform and hence the name Fourier damping. In practice as we approach the pole the amount of damping on any particular wave number increases and so the procedure reduces the model's ability to resolve features in the λ -direction. This is equivalent to degrading the resolution in that direction or alternatively to increasing the effective grid-length. We can thus consider this approach as a way of obtaining almost uniform resolution in meters rather than in radians. The Fourier damping procedure is as follows;

For each row ϕ find the largest value of $k \leq n/2$, where n is the number of points

on a row, such that the following is satisfied

$$\max_{\lambda} |u(\lambda, \phi)| \frac{\Delta t}{a \Delta \lambda \cos \phi} \sin(k \Delta \lambda) < 1$$

If $k = n/2$ than there are no unstable waves on that row. Experiments have shown that it is beneficial to insist that no row polewards of another can have more stable waves than that row. For each row with unstable waves we perform a discrete fast Fourier transform, often abbreviated to fft, to obtain the real and imaginary Fourier coefficients for each wave-number. We now damp the unstable waves by setting

$$r_j^d = r_j \left(\frac{k}{j}\right)^2 \quad k < j \leq \frac{n}{2}$$

where k is the maximum stable wave number, j is the wave number we wish to damp and r_j and r_j^d represent the undamped and damped Fourier coefficient respectively. Both real and imaginary coefficients are damped equally. We now perform another fast Fourier transform to return the damped wave representation to grid-point values. The choice of damping function is empirical and we have chosen the one used in the U.K. Meteorological Office unified model, see Cullen et al [19].

The equations at the poles and the equator

In spherical geometry we need to define what we mean by the equations at the poles. The mass equation becomes,

$$h^{np} = -D^{np} = -\sum \delta_{\phi c} V = \sum_{\lambda} \frac{1}{\cos^{np} \phi} V(\lambda, \pi/2 - \Delta\phi/2) \cos(\pi/2 - \Delta\phi/2)$$

$$h^{sp} = -D^{sp} = -\sum \delta_{\phi c} V = -\sum_{\lambda} \frac{1}{\cos^{sp} \phi} V(\lambda, -\pi/2 + \Delta\phi/2) \cos(-\pi/2 + \Delta\phi/2)$$

where the superscripts np and sp denote the values at the north and south poles respectively. The U term does not appear since the integral of $\delta_{\lambda} U$ around a row

is identically zero. For the momentum equations we need only consider u_0 since we defined v_0 at the poles and equator in section 3.2.1. On the u_0 row next to the pole we need to define U at the pole to use in the calculation of \overline{U}^ϕ ; we set U at the pole to zero. For the rows half a grid-length either side of the equator we have to form \overline{U}^ϕ and so we require U on the equator and this is not well defined. We therefore define U on these rows to be the value of U on the row half a grid-length polewards of this row. This choice gives a significant reduction in numerical noise near the equator when compared to using the value of U at the equator in forming U half a grid-length either side of the equator. We note that across the equator the inertial stability code, to be described in section 3.2.3, and used to enforce ellipticity and hence the existence of a solution, ensures that

$$u_0(\lambda, \Delta\phi/2) = u_0(\lambda, -\Delta\phi/2) \forall \lambda$$

which has the effect of removing the cross equatorial advection of u_0 since $V\delta_\phi u_0$ is identically zero across the equator.

B. The Semi-Lagrangian Scheme

Consider the forced advection equation for some scalar P

$$\frac{DP}{Dt} = F(P, x, y, t)$$

where x and y are space variables, t is time and F some known forcing function. Given this equation at time t_0 the solution at time t_1 can be obtained via a *Lagrangian* approach as

$$P_a = P_d + \int_d^a F$$

where subscript a denotes the value at the arrival point at time t_1 , d the value at the departure point at time t_0 and the integral is taken along the trajectory followed. Suppose we solve this equation via a fully Lagrangian approach on a time interval $[t_0, t_1]$ with the departure points at time t_0 equally spaced. At time t_1

the arrival points will generally have evolved to be highly irregularly spaced, see for example Welander [75]. The way to avoid this is to use the *semi-Lagrangian* approach. Now instead of fixing the departure points as equally spaced we require instead that the arrival points are equally spaced. The arrival points are now known and we must calculate the departure point, and since this will generally not be a point where the value of X is known we will need to interpolate to this point. To finish a discretised approach we need only approximate the integral along the trajectory. We consider each of these three elements in turn and state accuracy and stability results. In the implementation used here we follow the two-time-level approach of McDonald and Bates [44]. A review of the semi-Lagrangian method can be found in Staniforth and Côté [72] where they also discuss three-time-level schemes.

Departure point calculation

We wish to find the co-ordinates of the departure point (x_d, y_d) at time t for the trajectory which arrives at (x_a, y_a) at time $t + \Delta t$. If we apply the mid-point rule we obtain

$$x_d = x_a - \frac{\Delta t}{\Delta x} u_m(t + \Delta t/2)$$

$$y_d = y_a - \frac{\Delta t}{\Delta y} v_m(t + \Delta t/2)$$

where subscript m denotes the value at the mid-point of the trajectory. McDonald [42] used the arrival point values of \underline{u} at time t to approximate $\underline{u}_m(t + \Delta t/2)$ and successfully integrated the shallow-water primitive equations. This approach is only first order accurate in time and McDonald and Bates [43] showed that a second order accurate calculation gives noticeable benefits for timesteps greater than 20 minutes, which we also verify in chapter 4. Robert [61] argued that an improved estimate of the departure point can be obtained by improving the

estimate of \underline{u}_m . This is done by solving the iterative equations

$$\underline{u}_m^n(t + \Delta t/2) = \underline{u}(x_a - u_m^{n-1} \Delta t / \Delta x, y_a - v_m^{n-1} \Delta t / \Delta y, t + \Delta t/2)$$

where superscript n denotes the n th iterate and the first guess is obtained by extrapolating \underline{u} at time t at the arrival points to time $t + \Delta t/2$ using a second order accurate extrapolation. For example,

$$\underline{u}(t + \Delta t/2) = (3/2)\underline{u}(t) - (1/2)\underline{u}(t - \Delta t)$$

which gives a second order accurate temporal approximation to the departure point. In practice McDonald [42] found that a single iteration with a bi-linear interpolation to obtain the value at the mid-point was sufficient, with more iterations and higher order interpolation yielding no noticeable benefit. In chapter 4 we shall use both first and second order accurate trajectory routines.

Interpolation

The spatial accuracy of the advective part of the scheme is determined by the accuracy of the interpolation of the quantity at the departure point. The spatial truncation error is found to be one order higher than the order of the interpolation so that, for example, cubic interpolation gives a fourth order accurate scheme, see for example Staniforth and Côté [72]. A second order accurate scheme can be obtained by linear interpolation but many authors, eg McDonald [42], have found such schemes to cause unacceptably large damping of the solution. Following most authors we use bi-cubic interpolation and hence a fourth order accurate advection scheme is obtained.

For the unforced advection problem this combination of trajectory calculation and interpolation is unconditionally stable, see Staniforth and Côté [72] for references.

This just leaves the forcing term to consider. A second order temporal accurate approximation to the forcing term is given by the trapezium rule

$$(F(t + \Delta t) + F(t))/2$$

and thus the whole scheme is second order accurate in time, fourth order in space.

Implementation in the SSG model

The mass equation written in Lagrangian form is

$$\frac{Dh}{Dt} = -h\nabla \cdot \underline{u}$$

which becomes

$$h_a = h_d - \Delta t(\nabla \cdot \underline{U}_d(t) + \nabla \cdot \underline{U}_a(t + \Delta t))/2$$

when discretised using a semi-Lagrangian approach and the trapezium rule. The first problem is that we do not know \underline{U} at $t + \Delta t$ and we cannot calculate it until after the correction step. An alternative second order accurate form is to use the mid-point rule to obtain

$$h_a = h_d - \Delta t \nabla \cdot \underline{U}_m(t + \Delta t/2)$$

where we could find \underline{U} at the mid-point using the same calculation that we used to find \underline{u} for the calculation of the departure point. The more significant problem is the same as when we considered using the Heun scheme to solve this equation and is the inconsistency introduced between how the equation is solved in the corrector and predictor steps. This inconsistency, as noted earlier, can lead to large errors near the equator where only $\nabla \cdot \underline{U}$ is well defined and \underline{u} is not. We therefore use the Eulerian form of the mass equation exactly as we did in the Heun scheme

$$h_a(t + \Delta t) = h_a(t) - \nabla \cdot \underline{U}_a(t)$$

and this gives only first order accuracy in time, second order accuracy in space. For momentum we adopt the vector approach of McDonald and Bates [44] as described in Bates et al. [7]. We write the equation in its vector form

$$\frac{D\underline{u}_0}{Dt} = -g\nabla h - f\underline{k} \times \hat{\underline{u}}$$

where \underline{k} is the usual unit vector in the direction normal to the surface of the sphere and

$$\hat{\underline{u}} = \left(\frac{U}{h^\lambda}, \frac{V}{h^\phi} \right)$$

This can be discretised as

$$\begin{aligned} \underline{u}_{0_a} = \underline{u}_{0_d} - \frac{\Delta t}{2} [(g\nabla h_d(t) + (f\underline{k} \times \hat{\underline{u}}(t))_d) \\ + (g\nabla h_a(t + \Delta t) + (f\underline{k} \times \hat{\underline{u}}(t + \Delta t))_a)] \end{aligned} \quad (3.22)$$

to give second order accurate solutions in space and time. The vector approach is described in detail in Bates et al. [7] and we omit the details here. We note that this approach has no problems with the polar singularity and any velocities required at the poles can be obtained by finding the correct local cartesian vector at the pole via the method described at the end of section 3.2.1. At the equator U is defined as the average of the U velocities either side of the equator. The inertial stability enforcing conditions constrain u_0 to be constant across the equator, see section 3.2.3, and in an Eulerian scheme, like the Heun scheme, this stops cross-equatorial transport of \underline{u}_0 . In the semi-Lagrangian scheme it is found that using V on the row next to the equator in the calculation of the trajectories leads to significant numerical noise, which degrades the solution. If we set V on the row either side of the equator to zero the noise problem is removed. For typical timesteps, an hour or less, this stops trajectories from crossing the equator and so we are again stopping cross-equatorial transport of \underline{u}_0 . In neither code is this a major restriction since near the equator \underline{u}_0 is approximately zero and it is the cross-equatorial mass transport that is important. The correction step will adjust

the balanced velocity field to be consistent with the mass field and so this extra restriction, imposed to reduce numerical noise, is not a serious one.

This leaves us with the forcing terms to consider. As already noted we do not have a value for \underline{U} at time $t + \Delta t$ and so the form in equation (3.22) is not appropriate. As we considered with the mass equation we could replace this approximation with a mid-point value and still retain second order accuracy. If we use the mid-point rule for just the term involving \underline{U} then we are now evaluating the two forcing terms at different points and time-levels which would lead to larger errors for geostrophically balanced flows where these two terms should cancel. To minimise these errors we should only use the values at time t since here the flow is balanced whilst at time $t + \Delta t$, we have yet to compute the exact balanced flow and we have only estimates of the values. We thus arrive at the first order accurate in time discretisation

$$\underline{u}_{0_a} = \underline{u}_{0_d} - \Delta t [g \nabla h_d(t) + (f \underline{k} \times \hat{\underline{u}}(t))_d] \quad (3.23)$$

Implementing this discretisation leads to model runs failing in the region of the poles where the constraint enforcement procedure described in section 3.2.3 fails to remove areas where negative q appears and hence the correction equation ceases to be elliptic. The problem is the appearance of the negative values of q . The reason for their appearance in the solution can be found by looking at the derivation of the evolution equation for q , see section 2.2.2. Near the poles $\partial f / \partial \phi$ is approximately zero and q should be almost conserved. To obtain this conservation we noted in section 2.2.2 that we require

$$\frac{\partial}{\partial x} \left(\frac{\partial h}{\partial y} \right) = \frac{\partial}{\partial y} \left(\frac{\partial h}{\partial x} \right)$$

which in spherical polars becomes

$$\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{1}{a} \frac{\partial h}{\partial \phi} \right) = \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{\cos \phi}{a \cos \phi} \frac{\partial h}{\partial \lambda} \right)$$

Using the discretisation of equation (3.23) we find that this does not hold at the departure points since they may no longer be uniformly spaced due to both the evolution of the trajectories for the different points on the staggered grid and errors in calculating the trajectories. This requirement can be satisfied by using the values at the arrival point at the old time-level which is still a first order accurate in time approximation given by

$$\underline{u}_{0_a} = \underline{u}_{0_d} - \Delta t [g\nabla h_a(t) + (f\mathbf{k} \times \hat{\mathbf{u}}(t))_a] \quad (3.24)$$

Using this discretisation we find that the code now executes without any sign of developing areas of negative q and with no obvious detriment to the solution.

We note that both prediction schemes, Heun and semi-Lagrangian, use forcing terms and mass equation which are first order accurate in time. The only formal differences are that the semi-Lagrangian scheme as implemented is fourth order accurate in space for the unforced advection of \underline{u}_0 compared to second order accuracy in the Heun scheme. It is relatively easy to code a fourth order accurate version of the Heun scheme, see for example Cullen and Davies [18], but here we have left it as second order. Since we are not using semi-Lagrangian advection for all the quantities we cannot show that the coupled system is unconditionally stable and we expect that there may be a limit on the timestep we can take. This compares with the Heun scheme where we know that it is unconditionally unstable in theory and we probably have to rely on the correction step to keep the scheme stable.

3.2.3 Constraint Enforcement

In section 2.3 we defined the solution set S in which a solution exists. In the numerical procedure it is possible for the fields obtained at the end of the prediction step to lie outside this set at certain points. We wish to put these points back into the solution set whilst making the minimum changes to the fields H and

u_0 and here we consider how we might do this. In section 2.3 we identified two processes which apply in the f -plane case for $f \neq 0$ and $f = 0$ respectively. In the first case the solution can always be found by re-arranging the fluid to satisfy the constraint and in the second the only solution is $h = \text{constant}$ which can be obtained by uniformly mixing all the fluid. In the numerical code we must ensure that the partial differential equation to be solved in the correction step is elliptic, and this is the analogue of the existence conditions for the theoretical problem. We approximate the first process by simply re-arranging the balanced velocities to satisfy the inertial stability conditions. The conditions are

$$f(f + \delta_\lambda v_0 + u_0 \tan \phi / a) > 0 \quad (3.25)$$

$$f(f - \delta_\phi u_0) > 0 \quad (3.26)$$

We note that the first condition depends on u_0 whilst the second is independent of v_0 . We enforce the second condition first and then use the resulting u_0 field in enforcing the first condition. The conditions are enforced via an iterative procedure. At the start of each iteration of the routine to enforce the second condition we set u_0 on the row either side to the mean value of the current values on those rows. We then form

$$E = f(f - \delta_\phi u_0^r)$$

and if $E \leq 0$ at some point (λ, ϕ) we set

$$u_0^{r+1}(\lambda, \phi + \Delta\phi/2) = u_0^r(\lambda, \phi + \Delta\phi/2) + \alpha E / f - m$$

$$u_0^{r+1}(\lambda, \phi - \Delta\phi/2) = u_0^r(\lambda, \phi - \Delta\phi/2) - \alpha E / f + m$$

where superscript r denotes the r th iteration with the initial data values corresponding to those at $r = 0$, α is a parameter set to 0.7 which was found by experiment to give the best convergence rate and m is a minimum increment to be added, set to 1.25×10^{-3} , which was also found to be beneficial in speeding up

the iteration near the equator. If, on iteration r , no updates are performed then the condition is enforced. A solution always exists since u_0 set to the global mean of the input values at the start of the routine, satisfies the algorithm. This routine mimics a simple arrangement of the parcels by swapping values whilst also including some smoothing which is comparable to the mixing process required in the $f = 0$ case.

The first condition can be enforced on each row as it has no global dependence, unlike the u_0 condition where the mean value at the equator couples all the columns together. We set v_0 at the equator to zero and then at each point on the other rows we form

$$E = f((f + \delta_\lambda v_0^n + u_0 \tan \phi / a)$$

where u_0 is the value obtained after enforcing the second inertial stability condition, (3.26). If $E \leq 0$ at some point (λ, ϕ) we form the updates

$$v_0^{n+1}(\lambda + \Delta\lambda/2, \phi) = v_0^n(\lambda + \Delta\lambda/2, \phi) - \alpha E / f + m$$

$$v_0^{n+1}(\lambda - \Delta\lambda/2, \phi) = v_0^n(\lambda - \Delta\lambda/2, \phi) + \alpha E / f - m$$

where α and m have the same values as before. A solution always exists provided that $u_0 \tan \phi / a$ is not significantly greater than f , which is the case only for non-physical flows by which time the solution is meaningless.

In the development of the code the inability to remove inertial instability was the usual sign that the code was failing and this was used to determine when the procedure should be stopped. Enforcing inertial stability is not sufficient to guarantee existence of the solutions, see Shutts and Cullen [69], for the exact conditions. Instead we require that q is greater than some positive, data dependent, value which, as we shall see in the next section, is sufficient to guarantee ellipticity of the reduced elliptic equation (3.37). However, solving the reduced equation with inertial stability enforced and the full ellipticity condition not satisfied is found to still lead to the generation of spurious solutions. We must therefore find

a way of ensuring that the full condition is satisfied. It has not been possible to find a 2-D finite difference discretisation that mimics the f -plane re-arrangement procedure performed by the geometric model of Cullen and Purser [21], (*Cullen personal communication*). We therefore implement a global mixing procedure which is done using a diffusion scheme. This scheme is applied to $\tilde{q} = hq$ rather than q as it is computationally cheaper to implement and numerical experiments showed that it had no detrimental impact. The first step is to form \tilde{q} and then solve the following conservative diffusion equation for the increments $\Delta\tilde{q}$

$$\Delta\tilde{q} = \frac{1}{a\cos\phi} \frac{\partial}{\partial\lambda} \left(\frac{C_\lambda}{a\cos\phi} \frac{\partial\tilde{q}}{\partial\lambda} \right) + \frac{1}{a} \frac{\partial}{\partial\phi} \left(C_\phi \frac{1}{a} \frac{\partial\tilde{q}}{\partial\phi} \right)$$

where we have to specify the coefficients C_λ and C_ϕ . We wish to diffuse \tilde{q} only in the areas where it is likely to violate the ellipticity conditions and so we choose to specify the coefficients dependent on the current value of \tilde{q} relative to the value \tilde{q} would have if there was no flow, namely f^2 . We first note that the maximum stable diffusion coefficients given by linear stability analysis are

$$C_\lambda^{max}(\phi) = \frac{1}{4} \frac{(a\Delta\lambda\cos\phi)^2}{\Delta t}$$

$$C_\phi^{max} = \frac{1}{4} \frac{(a\Delta\phi)^2}{\Delta t}$$

where the λ -direction coefficient depends on the latitude ϕ . We now define the \tilde{q} dependent coefficients at a point (λ, ϕ) as

$$C_\lambda(\lambda, \phi) = \begin{cases} 0 & \text{if } \tilde{q} > \alpha f^2 \\ C_\lambda^{max}(\phi) \times (\alpha f^2 - \tilde{q}) / (f^2(\alpha - \beta)) & \text{if } \alpha f^2 \geq \tilde{q} \geq \beta f^2 \\ C_\lambda^{max}(\phi) & \text{otherwise} \end{cases}$$

and

$$C_\phi(\lambda, \phi) = \begin{cases} 0 & \text{if } \tilde{q} > \alpha f^2 \\ C_\phi^{max} \times (\alpha f^2 - \tilde{q}) / (f^2(\alpha - \beta)) & \text{if } \alpha f^2 \geq \tilde{q} \geq \beta f^2 \\ C_\phi^{max} & \text{otherwise} \end{cases}$$

where α and β are parameters which control the switching on of the diffusion and the point below which the maximum stable value should be used. In practice we found that

$$\alpha = .25, \beta = .15$$

gave good results for all the cases considered in the next chapter. We now need to find the implied increments to the height field given by the diffusive correction $\Delta\tilde{q}$ since we can then find the corresponding increments to \underline{u}_0 via the balance equations. We need to solve

$$\begin{aligned} & \left[f + \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f \cos \phi} \frac{\partial(h + \Delta h)}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial(h + \Delta h)}{\partial \phi} \right] \\ & \quad \left[f + \frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f} \frac{\partial(h + \Delta h)}{\partial \phi} \right) \right] \\ & \quad - \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f \cos \phi} \frac{\partial(h + \Delta h)}{\partial \lambda} \right) \right] \\ & \quad \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f} \frac{\partial(h + \Delta h)}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial(h + \Delta h)}{\partial \lambda} \right] = \tilde{q} + \Delta\tilde{q} \end{aligned}$$

This simplifies to

$$\begin{aligned} & \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial(\Delta h)}{\partial \phi} \right] \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f} \frac{\partial(\Delta h)}{\partial \phi} \right) \right] \\ & - \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right) \right] \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f} \frac{\partial(\Delta h)}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right] \\ & \quad + M_x \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f} \frac{\partial(\Delta h)}{\partial \phi} \right) \right] \\ & \quad + N_y \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial(\Delta h)}{\partial \phi} \right] \\ & \quad - M_y \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{a f} \frac{\partial(\Delta h)}{\partial \phi} \right) + \frac{g \tan \phi}{a^2 f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right] \\ & \quad - N_x \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{a f \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right) \right] = \Delta\tilde{q} \end{aligned}$$

where we have used the notation defined in section 2.2.3, page 28. This is a very complicated and expensive equation to solve for the correction, considering that we have chosen the coefficients using tunable parameters. We therefore wish to

simplify it to reduce the computational cost involved. We begin by neglecting the terms in Δh^2 which removes the first two terms and leaves us with a variable coefficient second order partial differential equation for Δh . This new equation has an ellipticity condition which depends on \tilde{q} . To avoid having to enforce this condition we can further simplify the equation by neglecting the mixed derivative terms which produces the more strongly elliptic equation

$$M_x \left[\frac{1}{a} \frac{\partial}{\partial \phi} \left(\frac{g}{af} \frac{\partial(\Delta h)}{\partial \phi} \right) \right] + N_y \left[\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{g}{af \cos \phi} \frac{\partial(\Delta h)}{\partial \lambda} \right) - \frac{g \tan \phi}{a^2 f} \frac{\partial(\Delta h)}{\partial \phi} \right] = \Delta \tilde{q}$$

This is elliptic if we have M_x and $N_y > 0$ which is simply the condition that the data is inertially stable. If we consider f to be constant then we could further simplify the equation by noting that typically

$$M_x/f \approx 1, N_y/f \approx 1$$

to obtain

$$g \nabla^2 (\Delta h) = \Delta \tilde{q} \tag{3.27}$$

We choose to solve equation (3.27) rather than the full equation since it is computationally very cheap to do so, and in test cases there was no noticeable benefit to the solution from solving the full equation. Solving this equation to obtain the corrections works well in the examples we have tried and alleviates any problems with the formation of areas of negative q .

3.2.4 Correction Step

We follow the argument of section 3.1.2 which described how the corrector step could be constructed in a theoretical manner except that now we consider how this may be implemented on the C -grid. We use a simple explicit Euler advection step to discretise the momentum equation and obtain the C -grid analogue of

equations (3.6) and (3.8)

$$\Delta h = -(\delta_\lambda \Delta U + \delta_{\phi c} \Delta V) \quad (3.28)$$

$$\Delta u_0 = -\left(\overline{\left(\frac{\Delta U}{h^\lambda}\right)^\phi} \delta_\lambda u_0\right)^\lambda - \frac{\Delta V}{h^\phi} \delta_{2\phi} u_0 + \left(\overline{\frac{\Delta U}{h^\lambda} v_0}\right)^{\lambda\phi} \frac{\tan\phi}{a} - g\delta_\lambda \overline{\Delta h}^\phi + \frac{f\Delta V}{h^\phi} \quad (3.29)$$

$$\Delta v_0 = -\frac{\Delta U}{h^\lambda} \delta_{2\lambda} v_0 - \left(\overline{\left(\frac{\Delta V}{h^\phi}\right)^\lambda} \delta_\phi v_0\right)^\phi - \frac{\Delta U}{h^\lambda} \overline{u_0}^{\lambda\phi} \frac{\tan\phi}{a} - g\delta_\phi \overline{\Delta h}^\lambda - \frac{f\Delta U}{h^\lambda} \quad (3.30)$$

where Δ represents the difference between the true values at the new time level and the current guess, which in the case of h and $\underline{u_0}$ is the predicted value at the new time level whilst for \underline{U} it is the value at the old time-level. We calculate the residuals

$$R_{ew} = g\delta_\lambda h - f v_0$$

$$R_{ns} = g\delta_\phi h - f u_0$$

and then form the correction equations

$$g\delta_\lambda \Delta h - f \Delta v_0 = -R_{ew} \quad (3.31)$$

$$g\delta_\phi \Delta h + f \Delta u_0 = -R_{ns} \quad (3.32)$$

We can now substitute for the corrections Δh and $\Delta \underline{u_0}$ from equations (3.28)–(3.30) in equations (3.31) and (3.32) and noting that $\Delta h = -\Delta D$ we obtain the discretised form of equations (3.11) and (3.12)

$$g\delta_\lambda \Delta D - f \left[f + \frac{1}{h^\lambda} \delta_{2\lambda} v_0 + \frac{1}{h^\lambda} \overline{u_0}^{\lambda\phi} \frac{\tan\phi}{a} \right] \Delta U - f \left(\overline{\left(\frac{\Delta V}{h^\phi}\right)^\lambda} \delta_\phi v_0 \right)^\phi + f g \delta_\phi \overline{\Delta D}^\lambda = R_{ew} \quad (3.33)$$

$$g\delta_\phi \Delta D + f \left[\left(\overline{\left(\frac{\Delta U}{h^\lambda}\right)^\phi} \delta_\lambda u_0 \right)^\lambda - \left(\overline{\frac{\Delta U}{h^\lambda} v_0} \right)^{\lambda\phi} \frac{\tan\phi}{a} \right] - f \left[f - \frac{1}{h^\phi} \delta_{2\phi} u_0 \right] \Delta V - f g \delta_\lambda \overline{\Delta D}^\phi = R_{ns} \quad (3.34)$$

If we were to continue following the procedure of section 3.1.2 we would now need to substitute for ΔV in equation (3.33) using equation (3.34) and for ΔU in equation (3.34) from equation (3.33). However because of the averaging incurred by using the staggered grid this would cause the equations at any point to be coupled to the equations at all other points. The effect of the coupling decreases with distance from the point about which the original substitution is performed. However it is not practical to pursue this line and so we seek to simplify the equation accepting the penalty that we shall not be able to remove all the residual in one go. This is similar to omitting the term in the SIMPLEC algorithm which leads to a simpler scheme at the expense of extra iterations, see Van et al. [73] and Wen et al. [76]. We require that the iteration we construct must converge to the same solution as that obtained for the full equation, and this has yet to be proven. We can avoid the coupling if we neglect the terms involving ΔV in equation (3.33) and ΔU in equation (3.34). If we also neglect the ϕ derivative of ΔD in equation (3.33) and the λ derivative of ΔD in equation (3.34) since these terms are order f smaller than the derivatives of ΔD in the other co-ordinate direction in the respective equations, then we obtain the reduced equations

$$g\delta_\lambda\Delta D - f \left[f + \frac{1}{h^\lambda}\delta_{2\lambda}v_0 + \frac{1}{h^\lambda}\bar{u}_0^{\lambda\phi}\frac{\tan\phi}{a} \right] \Delta U = R_{ew} \quad (3.35)$$

$$g\delta_\phi\Delta D - f \left[f - \frac{1}{h^\phi}\delta_{2\phi}u_0 \right] \Delta V = R_{ns} \quad (3.36)$$

For simplicity we define

$$A = \frac{1}{f \left[f + \frac{1}{h^\lambda}\delta_{2\lambda}v_0 + \frac{1}{h^\lambda}\bar{u}_0^{\lambda\phi}\frac{\tan\phi}{a} \right]}$$

$$B = \frac{1}{f \left[f - \frac{1}{h^\phi}\delta_{2\phi}u_0 \right]}$$

Then taking the divergence of equations (3.35) and (3.36) we obtain away from the equator

$$\delta_\lambda(gA\delta_\lambda\Delta D) + \delta_{\phi c}(gB\delta_\phi\Delta D) - \Delta D = \delta_\lambda(AR_{ew}) + \delta_{\phi c}(BR_{ns}) \quad (3.37)$$

which is elliptic if the data is inertially stable, and we have shown in section 3.2.3 that this is easily enforced. This ellipticity condition is much simpler than the one derived in section 3.1.2 and is a necessary but not sufficient condition for the full ellipticity condition to hold. We now need to derive the equation at the equator. In section 3.1.2 we noted that if we enforce inertial stability on the data then we have removed the residuals on the equator. When we enforce inertial stability we do so only on the balanced velocities and not on the height field. This is because of the difficulty in enforcing

$$f \left[f + \delta_\lambda \left(\frac{g}{f} \delta_\lambda h \right) + \overline{\left(\frac{g \tan \phi}{f a} \delta_\phi h \right)^\phi} \right] \geq 0$$

and

$$f \left[f + \delta_\phi \left(\frac{g}{f} \delta_\phi h \right) \right] \geq 0,$$

for which no effective numerical method could be found. Thus we are unable to continue to follow the approach of section 3.1.2. We note that an alternative to enforcing the Neumann boundary condition at the equator is to use a Dirichlet condition since all the results we have used for elliptic equations hold equally for this type of boundary condition. This leaves us with the problem of finding the solution at the equator before we can solve for the rest of the domain. The equation we need to satisfy at the equator is

$$\delta_\lambda \Delta D = R_{ew} \tag{3.38}$$

since the other balance equation is formed half a grid-length polewards of the equator where $f \neq 0$. (3.38) is easily solved to give

$$\Delta D = e(\lambda) + s(\phi)$$

where $e(\lambda)$ is the solution of equation (3.38) and $s(\phi)$ is some arbitrary function. We note that $s(\phi)$ is simply the zonal mean solution of equation (3.37), or the real

Fourier coefficient a_0 in a Fourier expansion of the solution ΔD in the λ -direction given by

$$\Delta D(\lambda, \phi) = a_0(\phi) + \sum_{k=1}^{\infty} a_k(\phi) \cos(k\lambda) + ib_k(\phi) \sin(k\lambda)$$

We could now use this transform on equation (3.37) to find the coefficient a_0 . A cheaper, although exactly equivalent, way is to integrate both sides of the equation with respect to λ over $[0, 2\pi)$. Doing this we obtain,

$$\int_0^{2\pi} [\delta_\lambda(gA\delta_\lambda\Delta D) + \delta_{\phi_c}(gB\delta_\phi\Delta D) - \Delta D] d\lambda = \int_0^{2\pi} [\delta_\lambda(AR_{ew}) + \delta_{\phi_c}(BR_{ns})] d\lambda$$

and noting that

$$\int_0^{2\pi} \Delta D d\lambda = a_0$$

we can write this as

$$\begin{aligned} \int_0^{2\pi} gB\delta_{\phi_c}(\delta_\phi\Delta D) d\lambda - a_0 &= \int_0^{2\pi} [\delta_\lambda(AR_{ew}) + \delta_{\phi_c}(BR_{ns})] \\ &\quad - \int_0^{2\pi} [\delta_\lambda(gA\delta_\lambda\Delta D) + \delta_{\phi_c}(gB\delta_\phi\Delta D)] d\lambda \end{aligned}$$

If $\delta_\lambda A = 0$ then the first term in the second integral on the right-hand-side is identically zero. If we also assume that the second term in this integral is negligible and approximate

$$\int_0^{2\pi} gB\delta_{\phi_c}(\delta_\phi\Delta D) d\lambda$$

by

$$\delta_{\phi_c}(\delta_\phi a_0) \int_0^{2\pi} gB d\lambda$$

we obtain

$$\delta_{\phi_c}(\delta_\phi a_0) \int_0^{2\pi} gB d\lambda - a_0 = \int_0^{2\pi} [\delta_\lambda(AR_{ew}) + \delta_{\phi_c}(BR_{ns})] d\lambda$$

The solution of this is easily obtained since on the C-grid B is not evaluated at $f = 0$ and hence does not become infinite. The omission of terms and the approximations made imply some doubt about the accuracy of the solution obtained by this method. However, the approximations made are no worse than

those made to derive the reduced equation (3.37) and at worst should just add extra iterations to the procedure. Using this method to find a_0 and then solving equation (3.37) with the Dirichlet boundary condition does produce a noticeable unremoved residual at the equator in the multigrid solution on the first application of the corrector step on a timestep, which is removed on the subsequent iteration of the corrector step. No adverse effects appear from this unremoved residual and idealised experiments with the multigrid code suggest that the error may be entirely due to the multigrid algorithm, and not because of the simplifications made here, see section 3.3.4. The results gained via the solution of the reduced equation (3.37) and the Dirichlet boundary condition method are given in the next chapter, and as we saw in section 3.1.4 these simplifications do not appear to be detrimental to the solution in any way, except perhaps in a slight loss of accuracy. We discuss the solution of the variable coefficient elliptic equations on the sphere in section 3.3. Equation (3.37) is solved via the hemispheric method described there, whilst the solution to Poisson's equation required in the constraint enforcement is solved using the global multigrid code.

Having solved the *reduced* equation (3.37), then we can find ΔU and ΔV uniquely by substituting ΔD in equations (3.35) and (3.36), with the exception that we cannot find ΔU at the equator. This problem is overcome by noting that both ΔD at the equator and ΔV either side of the equator are well defined and we can solve

$$\delta_\lambda \Delta U = \Delta D - \delta_{\phi_c} \Delta V$$

Solving this for ΔU at $\phi = 0$ given ΔD and ΔV does not give a unique solution as it introduces a constant of integration, in this case an arbitrary function of ϕ . This arbitrary function represents the rotational part of ΔU which we choose to set to zero. This is the point in the numerical procedure where the undefined part of the solution at the equator appears, in this case it is the rotational part of U which is undefined. This is no surprise since in the previous chapter we noted that

the rotational part of the wind field is undefined at the equator. We choose to set the rotational part of U to zero at the equator, and since it is not used anywhere else in the solution procedure its definition is of no importance. We can now use these corrections to calculate the corrections Δh and Δu_0 using equations (3.28) to (3.30). The value of \underline{U} at the new time level is given by $\underline{U} + \underline{\Delta U}/\Delta t$. By omitting the terms in the derivation of equation (3.37) we have sacrificed some accuracy to obtain the compact 5–point stencil and also the ability to remove all the residual in one corrector step since we are still using the full equations (3.28) to (3.30) to calculate the corrections. This forces us to perform some iterations of the correction procedure. In practice only one or two iterations are required to remove the residual.

3.2.5 The Complete Procedure

The full predictor/corrector scheme can be summarised as follows

Step 1 q dependent diffusion.

Step 2 Inertial stability enforced.

Step 3 Prediction step.

Step 4 Inertial stability enforced.

The following steps are iterated n times;

Step 5 Corrector step.

Step 6 Inertial stability enforced.

For some of the test problems it was found to be beneficial to include an inertial stability enforcing step after the diffusion step, particularly when the local timestepping Heun scheme was used as the predictor. The diffusion step is only performed once at the beginning of the timestep whilst the inertial stability test must be applied before every corrector step to ensure that the reduced equation to be solved is elliptic.

3.3 Multigrid Methods on the Sphere.

3.3.1 Introduction to Multigrid methods.

Many problems in computational mathematics require fast, accurate solutions to elliptic partial differential equations. If we consider the model problem,

$$\frac{\partial^2 Q}{\partial x^2} = f(x)$$

on a periodic domain, then if we discretise using, for example, centred finite-differences we obtain the matrix system,

$$A\underline{Q} = \underline{F}$$

where A is a square, symmetric tri-diagonal matrix. Fast direct methods exist for such a problem. If we now consider the more complicated equation,

$$a(x, y) \frac{\partial^2 Q}{\partial x^2} + b(x, y) \frac{\partial^2 Q}{\partial y^2} = f(x, y)$$

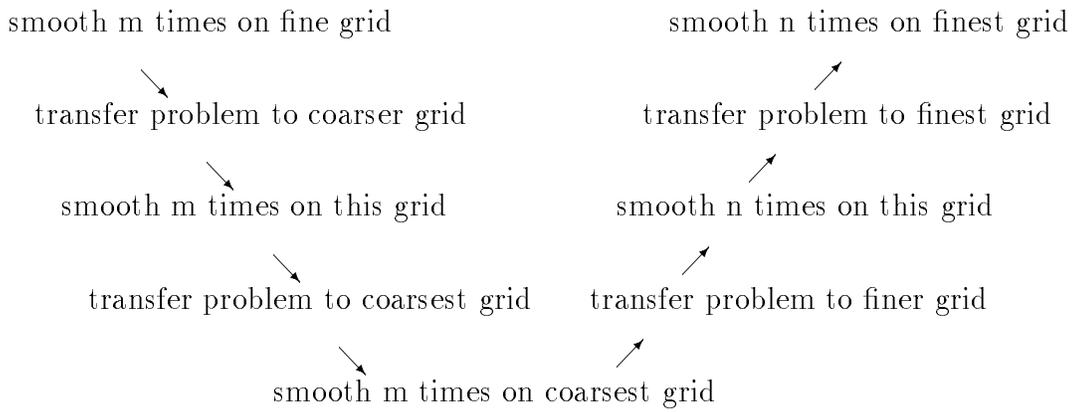
and again discretise we obtain a block tri-diagonal matrix system. For a $N \times N$ problem the cost of direct solution is $O(N^6)$ and takes $O(N^4)$ elements of memory storage. We can take advantage of the banded structure of this problem by using algorithms such as the Ahlberg-Nilsen-Walsh [1] which reduce costs to $O(N^4)$ operations. For certain classes of problems there are faster direct methods. For example if the coefficients a, b are simply functions of y then a Fourier transform can be employed in the x -direction to reduce the problem to simply solving a set of 1-D equations. This works in 2-dimensions but when we look at a 3-D problem then the cost of direct solution becomes totally prohibitive, $O(N^9)$. For general problems fast iterative methods have been developed which converge to machine precision. These iterative methods take, in most cases in 2-D, significantly less computer time than the direct method and require considerably less storage, while in 3-D they represent the only practical way of solving the problem. For elliptic

equations the *Multigrid* method has usually been shown to be the best iterative procedure, see Gary et al. [29] for example.

The basic strategy behind multigrid is to choose a computationally cheap solver for the matrix problem, referred to as the *smoother*, and combine it with a procedure which solves the problem on a sequence of computational grids. Given the matrix system $A\underline{x} = \underline{b}$ then a simple iterative solver is the *Jacobi* iteration where we separate A into three matrices, a diagonal, a strictly upper diagonal and a strictly lower diagonal, denoted D, U, L respectively and form the iteration,

$$D\underline{x}^{n+1} = -(L + U)\underline{x}^n + \underline{b}$$

Convergence for the Jacobi iteration is quite fast for a few iterations and then becomes very slow. The reason behind this is easily understood by looking at the Fourier modes of the error $\underline{e} = \underline{b} - A\underline{x}$. The Jacobi method converges quickly for the high frequency modes, often referred to as the ‘rough’ modes, of the problem but very slowly for the low frequency modes, often referred to as the ‘smooth’ modes. If we were to coarsen the problem by solving it on a grid with half as many nodes then the Jacobi method would now converge quickly for the ‘rough’ modes on this new coarser grid. We need to be able to perform this coarsening without changing the solution to the problem. This suggests the following procedure, here shown for just 3 grids.



The method shown above is called a (m,n) V-cycle multigrid method. The V comes naturally from the shape and the (m,n) describes the number of smoothing iterations performed on the descending and ascending parts of the cycle respectively. In many applications a direct solution method is employed on the coarsest grid to get an exact solution, however we use an iterative approach on this grid as well. As we are using an iterative approach on the coarse grid we shall define a (m,k,n) V-cycle in the same way but with k now referring to the number of iterations of the smoother on the coarsest grid. The transfer from a fine grid to a coarser grid is known as *restriction* whilst the transfer from coarse to finer is called *prolongation*. For most problems several V-cycles will be needed before the solution converges to the required level of accuracy.

Convergence of the multigrid procedure depends mainly on two things. Firstly the smoother chosen must smooth the rough modes on every grid. Secondly the restriction operator must be such that the problem on the coarser grid is the same as if the original problem had been posed on that grid. In practice it is found that this does not have to be exact but must however be close in some sense. A third, although lesser consideration, is that the prolongation operator is usually chosen to be the adjoint or inverse of the restriction operator. To obtain the

optimum scheme for any particular problem in practice it is usually necessary to investigate in some detail the performance of smoothers and various restriction and prolongation operators. Here we will limit ourselves to creating a robust method using the simple (m,k,n) V-cycle.

More background information can be found at a very basic level in Briggs [8], or for a more complete introduction see either of the books by Hackbusch [32] or Wesseling [77]. For a review of multigrid methods for typical elliptic problems which arise in meteorology see Fulton et al [28].

3.3.2 Solution of a class of second-order P.D.E.s on the sphere.

We wish to solve elliptic equations of the form,

$$\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} \left(\frac{A(\lambda, \phi)}{a \cos \phi} \frac{\partial Q}{\partial \lambda} \right) + \frac{1}{a \cos \phi} \frac{\partial}{\partial \phi} \left(\frac{B(\lambda, \phi) \cos \phi}{a} \frac{\partial Q}{\partial \phi} \right) - C(\lambda, \phi) Q = R(\lambda, \phi)$$

on the surface of a sphere radius a given that $C > 0$. The equations are elliptic provided that $A, B > 0, \forall \lambda, \phi$. The equations are written in spherical polar co-ordinates (λ, ϕ) where $\lambda \in [0, 2\pi)$ and $\phi \in [\pi/2, -\pi/2]$, so that $\phi = 0$ is the equator.

We also wish to consider the same problem everywhere except $\phi = 0$, with the solution Q given for all λ at $\phi = 0$, which is effectively the simultaneous solution of two hemispheric problems. To distinguish between them we shall call them the *Global* and *Hemispheric* problems respectively.

If the coefficients A, B, C were simply functions of ϕ , this form of equation could be solved very efficiently using a direct method, see for example Moorthi and Higgins [54]. However, this is not the case here and so we need a different approach. Barros [6] has applied multigrid methods to Poisson type equations on the sphere and this work is based on his.

We wish to solve the problem on an $M \times N$ equally spaced grid in (λ, ϕ) space where M is even and N is odd. We define $\Delta\lambda = 2\pi/M$ and $\Delta\phi = 2\pi/(N - 1)$. We denote by $Q_{i,j}$ the value of Q at $((i - 1)\Delta\lambda, \pi/2 - (j - 1)\Delta\phi)$. Thus there are M points at both north and south poles, although in practice we shall only need one. We discretise the problem using centred finite differences, so that,

$$\frac{\partial Q_{i,j}}{\partial \lambda} = (Q_{i+1/2,j} - Q_{i-1/2,j})/\Delta\lambda$$

with an equivalent definition for derivatives in the ϕ direction. If we locate the coefficients on the computational grid so as to avoid the need for any averaging then we arrive at the following stencil,

i	i+1/2	i+1	
Q	A	Q	j
B		B	j+1/2
Q	A	Q	j+1

If, for simplicity, we consider Poisson's equation, $\nabla^2\phi = f$, on the sphere, then using centred finite-differences we obtain a second order accurate solution. For the problems for which we wish to use multigrid this is sufficient. It is reasonably easy to implement more accurate schemes at the expense of increasing the size of the stencil and hence the minimum number of points required on the coarsest grid. Increasing the size of the coarsest grid increases the cost of direct solution on it. If an iterative method is used then in general the number of iterations required may increase quite dramatically, see for example the effect of one less

grid in the results in table 3.1 in section 3.3.4. This may also not be desirable if, as is required here, line solvers must be used as the smoother. This is because the matrices generated by the line solvers becoming increasingly complicated and more expensive to solve.

3.3.3 Implementation

Definition of convergence.

We need to define when convergence has occurred. There are two commonly used definitions which measure the performance of the multigrid scheme. We define the root mean square residual for the matrix problem $Ax = b$ as

$$rms = \left[\frac{1}{s} \sum_{i=1}^s (b_i - Ax_i)^2 \right]^{\frac{1}{2}}$$

where s is the number of elements of x . The first measure of convergence is the *Multigrid Convergence Rate* (henceforth MCR) which is defined simply as the root mean square residual at the end of a V-cycle divided by its value at the start of the V-cycle. The second measure is called *the Practical Smoothing Rate* (henceforth PSR) and is defined for a (m,n) V-cycle as,

$$(MCR)^{\frac{1}{m+n}}$$

The first measure is an absolute measure of convergence whilst the second reflects the amount of work done in the V-cycle and hence is a better measure for comparing the different values of (m,n) which could be used. As we are using (m,k,n) V-cycles we should probably take into account k; however as this is in practice little different from n or m we do not bother. We choose to define convergence as the point where the practical smoothing rate becomes close to 1, since when this happens we are getting very little extra convergence per V-cycle. In the

examples considered later we generally get a PSR value between .35 to .40 for all V-cycles except the last one or two. A good cutoff value for defining convergence has proved to be .950. This is because the scheme can often perform many more V-cycles with values around .97 to .99 before reaching a value of 1 but they gain us little extra benefit for often a substantial cost.

We note that it is not uncommon for certain problems for some iterative procedures to reach a plateau after a certain number of iterations. The scheme may remain there for many more iterations before once more converging quickly. This behaviour can be observed, for example, in conjugent gradient methods. We have not seen anyone publish results suggesting that this is also the case with multi-grid. For the test problems we have tried convergence has always been to machine precision with no evidence of a plateau. To confirm this we have re-run the same problems at greater precision on the computer and converged to this new value of machine precision. Thus we are reasonably confident that this convergence definition is useful.

Data representation at the pole.

The pole is simply one point although we hold M values there. However we keep all M polar values of each variable and each trigonometric function there the same to enable easier addressing in the computer code. The coefficients A are set to zero at the poles since there is no meaning to the λ directional derivative there and they are never used. The right-hand-side R is the value at the pole but is stored as the value averaged over all M points at the pole. Each point thus holds R/M . This is because when this value is used in the restriction by points away from the pole it is this average value which is required.

We note that the value of $\cos\phi$ applicable in any grid-box centred on (λ, ϕ) is actually the average value over the box with corners at $(\lambda \pm \Delta\lambda/2, \phi \pm \Delta\phi/2)$.

This was discussed in section 3.2.1, page 66. We note that as we change grids the value at the poles will change whilst the other values remain the same. However the polar cosine value can always be obtained simply as $\Delta\phi/4$ on any grid.

Grid sequence.

The grid sequence generated depends on whether we are solving the global or hemispheric problem.

In the λ -direction we require all grids to have an even number of points since this allows us to apply restriction by simply taking alternate grid points. This constraint could be removed but would require a more complicated restriction and hence is probably undesirable. Secondly the minimum number of points allowed in this direction is set to 4, as it takes three points to resolve the second order PDE in this direction. We could allow three points but this adds a little extra complexity to the code and experiments show that no extra benefits in terms of performance accrue.

In the ϕ direction we ensure that the number of points is always odd so that when we coarsen by taking alternate points in the ϕ -direction we still keep points at both poles and on the equator, which, besides keeping the restriction simple allows almost the same code to be used for both global and hemispheric problems. In the global case we set the minimum number of points to 5 rather than the 3 needed to resolve the problem. This is done for practical purposes only as experiments show we gain no extra efficiency from using just the 3 points. For the hemispheric problem the minimum number is set to 9 which is consistent with the global problem in that it has 3 points between the boundary conditions on the coarsest grid.

As an example of the grids generated we consider a 192×129 global problem where we restrict by taking alternate points in each direction every time.

Grid Index	λ	ϕ
6	192	129
5	96	65
4	48	33
3	24	17
2	12	9
1	6	5

Thus we end up with a 6×5 problem to solve on the coarsest grid, the grid generation being stopped by the ϕ -direction criteria becoming violated on the next grid down. For problems on the sphere, if we wish $\Delta\lambda$ to be equal to $\Delta\phi$ then we require $M = 2(N - 1)$. Thus, in general, the criteria for stopping generating grids will be violated first in the ϕ -direction. To enable more grids to be generated we allow restriction in just the λ -direction when the ϕ criteria becomes violated. For example here are the grids generated for the 192×129 hemispheric problem;

Grid Index	λ	ϕ
6	192	129
5	96	65
4	48	33
3	24	17
2	12	9
1	6	9

We see that the generation would have stopped on grid 2 since this is when the ϕ -direction criteria is violated but we now have allowed a sixth grid. The grid generation is now stopped on grid 1 as the λ -direction criteria is violated on the

next grid down. The reason behind numbering the coarsest grid 1 is that if we add more points then it is easier to see the structure growing.

Smoother.

We consider the problem $Ax = b$. We wish to smooth the rough modes of the error so we calculate the residual,

$$r = b - Ax_1$$

where x_1 denotes the current solution and solve the correction equation for Δx ,

$$A\Delta x = r$$

and then update the solution by

$$x_2 = x_1 + \Delta x$$

The question is thus which is the optimum iterative solver for the correction equation ? The solver must smooth the rough modes on all grids and be as computationally inexpensive as possible. The problem on the sphere is that we are not dealing with a square grid in cartesian space but with one with strong anisotropies. Consider again our example grid of 192×129 and calculate the ratio of $a\Delta\lambda\cos\phi$ to $a\Delta\phi$.

on Row 2 we get, 30.5 : 1

on Row 33 we get, 1 : 1

on Row 65 we get, 0.75 : 1

From this it is obvious that towards the poles the problem is more strongly coupled in the λ -direction than in the ϕ -direction and to a lesser extent the converse is true as we approach the equator. For problems exhibiting these strong anisotropies the usual point smoothers used in many multigrid algorithms have very poor

smoothing properties. However, linesmoothers do smooth very well, see for example Barros [6]. The disadvantage with line smoothers is that we now have to solve a 1-D matrix problem on each line and thus they are significantly more expensive per application than point smoothers. However because the smoothing rates for the line solver on an anisotropic problem are significantly better than those for the point solver, see Barros [6] for an example of typical smoothing rates, many point solves are required to reduce the residual by the same amount as one line solve. Thus for a strongly anisotropic problem to reduce the residual by a specified factor the line solver is usually computationally cheaper than the point solver. More information about point and line smoothers can be found in the books by Hackbusch [32] and Wesseling [77]. It would seem that we need only use a line solver in the λ -direction; however analysis, see Barros [6], also shows that this can be a poor smoother close to the equator due to the presence of the anisotropy in the ϕ -direction. The efficiency of the λ -direction solver decreases as the anisotropy in the ϕ -direction becomes stronger. The most robust solution is to combine line smoothers in both directions. This solution is expensive per V-cycle compared with the point smoothers but converges quickly. We need to investigate whether an alternating line relaxation procedure, where we first smooth in the λ -direction and then in the ϕ -direction, is required for the problems we are interested in or is a λ -direction line solver all that is required. There still remain two questions to answer. Firstly which type of line relaxation should we use ? Secondly how do we treat the poles ?

There are three common line relaxation schemes, Jacobi, Gauss-Seidel and Zebra. The Zebra scheme is the extension of the Red/Black point solver scheme to line solvers. Each line of the problem is coloured alternately red then black. We then solve on all the red lines and update the solution before solving on the black lines. The choice between the three line relaxation appears to be, for the problems considered here, an open question as they all have comparable performance in

terms of the number of V-cycles required. However the Zebra line relaxation offers the benefit of being vectorisable on a CRAY Y-MP computer and thus is significantly cheaper on this particular architecture, by a factor of between 6 and 8 times. The Zebra scheme is also well suited to parallel computer architectures, although we have yet to implement the scheme on a massively parallel computer. We consider how to implement multigrid on a massively parallel computer in section 3.3.5.

The poles are treated as follows. No λ -direction line solve is performed at the poles. In the ϕ -direction line solver, each line is solved for all the points not at the poles using the current polar values as Dirichlet boundary conditions. Then the polar values are updated using a point smoother, for example at the north pole;

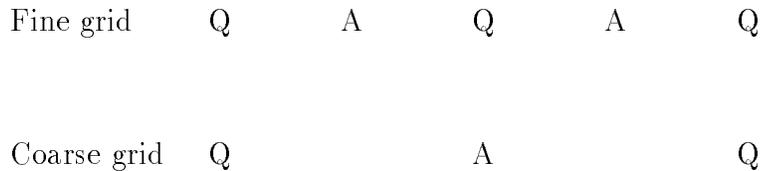
$$\left(-C + \sum_M \frac{B \cos(\pi/2 - \Delta\phi/2)}{\Delta y^2 \cos(\text{pole})}\right) \Delta x = b - A(x_1 + \Delta x)$$

where Δx on the right hand side is set to zero at the pole, and we note that we need to sum the residuals at each point at the pole to obtain b there due to the way we have chosen to store them. The residual calculated on the right hand side is the value at the end of the line solves. The cosine at the pole is not zero as noted earlier. This polar treatment makes the usual multigrid smoothing analysis difficult, however the procedure works very well in practice. We note that a recent paper by Douglas et al [26] has proposed a unified convergence theory for multigrid methods which avoids many of the previously required restrictions. These restrictions included A being symmetric which, in general, is not the case here. It may be possible to establish a result for the scheme proposed here using this theory. However, for problems with general coefficients, which are our goal, these estimates are probably only obtainable numerically.

Restriction.

We choose to perform the restriction by taking alternate points in each direction and we have to restrict the fields, A, B, C, Q, R and the trigonometric functions. Since the coefficients and the trigonometric functions are the same for every V-cycle we need only perform the restriction on them on the first cycle. To conserve certain properties of the original problem on the finest grid and also because of the grid staggering we need to apply the restriction differently to each field.

The trigonometric constants are simply injected and so remain exact. The exception to this is at the poles where we have to change the values. This is because, for example, the average over the grid box of $\cos\phi$ is always $\Delta\phi/4$ where $\Delta\phi$ is the grid length for the particular grid the scheme is on. Hence when we restrict we change $\Delta\phi$ and hence we must also change the trigonometric values at the poles. The solution Q and coefficient C are also injected but with no special treatment at the poles. The coefficient A cannot be transferred by injection since it is not held at the same point on the coarse grid as it was on the fine grid. This is easily seen from the following diagram showing the location of Q and A for both fine and coarse grids.



A on the coarse grid is calculated by linear interpolation from the coarse grid values. The B coefficient is treated similarly to the A one by using linear interpolation to obtain its value at the new grid location. This just leaves R to be restricted. We choose to restrict not R but the current residual, which we denote

by r , given by

$$r(\text{fine}) = R(\text{fine}) - LHS(\text{fine})$$

where LHS is the left hand side of our equation and the (fine) denotes that it is evaluated on the fine grid. After performing the restriction we then obtain the right hand side on the coarser grid by adding on the new value of the left hand side on the coarser grid,

$$R(\text{coarse}) = \text{restricted}(r(\text{fine})) + LHS(\text{coarse})$$

This process is known as the *full approximation scheme*, often abbreviated to *FAS*. We choose to perform the restriction using *full-weighting* not injection. Full-weighting in 1-D to a point i would look like

$$r_i(\text{coarse}) = (r_{i-1}(\text{fine}) + 2r_i(\text{fine}) + r_{i+1}(\text{fine}))/4$$

where the factor which is the divisor is simply the sum of the number of values inside the bracket. A simple way to generate full-weighting is to successively apply the averaging operator s defined by

$$s(r_i) = (r_{i-1/2} + r_{i+1/2})/2$$

until a desired full-weighting is obtained. Hence we could write the one defined here as

$$r_i(\text{coarse}) = s(s(r_i(\text{fine}))) \text{ or } s^2(r_i(\text{fine}))$$

The advantage of full-weighting is that it preserves the property that if

$$\sum_{\text{fine}} r \Delta x = 0$$

then

$$\sum_{\text{coarse}} r \Delta x = 0$$

The equivalent definition in 2-D on the sphere is a little more complicated since the area of the grid boxes is not constant. The required full-weighting of r at a

point i, j not at the pole is

$$\frac{1}{16\cos\phi_i} \begin{bmatrix} (r\cos\phi)_{i-1,j-1} & +2(r\cos\phi)_{i,j-1} & +(r\cos\phi)_{i+1,j-1} \\ +2(r\cos\phi)_{i-1,j} & +4(r\cos\phi)_{i,j} & +2(r\cos\phi)_{i+1,j} \\ +(r\cos\phi)_{i-1,j+1} & +2(r\cos\phi)_{i,j+1} & +(r\cos\phi)_{i+1,j+1} \end{bmatrix}$$

This is simply a grid box area weighted version of the 1-D full-weighting applied in 2-D. To complete the restriction we need the equivalent definition at the pole. The condition we must satisfy is, if

$$\sum_{fine} r\Delta\lambda\cos\phi\Delta\phi = 0$$

then

$$\sum_{coarse} r\Delta\lambda\cos\phi\Delta\phi = 0$$

where we sum over all points including every one at each pole. This is because of the way we chose to store the residual there. It is important to note, as pointed out in the section on data representation, that the polar cosine value changes as we change grids. Full-weighting at the pole, in this example the north pole, gives

$$\frac{r_{i,1}}{4} + \frac{1}{8\cos\phi_{i,1}} \sum_{i=1,M} r_{i,1}\cos\phi_{i,1}$$

where all the cosine values are those for the fine grid. As all the polar values are the same then we need only do this for 1 point and then set the others. Simple arithmetic then shows that we have satisfied the required condition.

The reason for requiring the restriction to satisfy this condition is that to solve Poisson's equation on the sphere the integral of the right hand side over the sphere must be zero. This is known as the *compatibility condition*. This condition must also be satisfied on all coarser grids as well and hence is the main reason for using full-weighting.

Prolongation.

The only field we need to transfer to a finer grid is the current value of the answer Q . This is done by linear interpolation since this it is cheap and also the inverse of the full-weighting restriction operator up to an arbitrary constant. The combination of linear interpolation and full-weighting restriction is probably the most commonly used in multigrid schemes. It is also probably the easiest to use in the standard multigrid smoothing analysis.

3.3.4 Results.

For the moment we choose to use a (2,3,2) V-cycle with an alternating zebra line relaxation scheme, where the numbers in the V-cycle definition refer to the number of alternating line solves on each grid not to the total number of line solves. Later in this section we consider how the choice of numbers in the V-cycle affects the results and also whether alternating line relaxation is required. The first problem we consider is simply that of Poisson's equation $\nabla^2 Q = F$ on the sphere. This equation possesses a unique solution only up to an arbitrary constant. This can easily be seen to be the case in the numerical solution since the centred finite form of Poisson's equation at a point (i, j) on the grid is

$$\frac{1}{a^2 \cos^2 \phi_{i,j} \Delta \lambda^2} (Q_{i+1,j} - 2Q_{i,j} + Q_{i-1,j}) + \frac{1}{a^2 \cos \phi_{i,j} \Delta \phi^2} \left((Q_{i,j+1} - Q_{i,j}) \cos \phi_{i,j+1/2} - (Q_{i,j} - Q_{i,j-1}) \cos \phi_{i,j-1/2} \right) = R_{i,j}$$

and $Q + c$ also satisfies this finite-difference equation for any constant c . However since we are using line solvers as our smoother we discover that the multigrid method can only have zero as its arbitrary constant. This is because the finite-

difference equation for the λ -direction line solver is

$$\frac{1}{a^2 \cos^2 \phi_{i,j} \Delta \lambda^2} (Q_{i+1,j} - 2Q_{i,j} + Q_{i-1,j}) - \frac{1}{a^2 \cos \phi_{i,j} \Delta \phi^2} (Q_{i,j} \cos \phi_{i,j+1/2} + Q_{i,j} \cos \phi_{i,j-1/2}) = R_{i,j}$$

and for the ϕ -direction line solver it is

$$\frac{-2Q_{i,j}}{a^2 \cos^2 \phi_{i,j} \Delta \lambda^2} + \frac{1}{a^2 \cos \phi_{i,j} \Delta \phi^2} ((Q_{i,j+1} - Q_{i,j}) \cos \phi_{i,j+1/2} - (Q_{i,j} - Q_{i,j-1}) \cos \phi_{i,j-1/2}) = R_{i,j}$$

and substituting $Q + c$ instead of Q into either of these forms gives $c = 0$ as the only constant which still satisfies the equation. Hence a nice benefit of using line-solvers in the numerical solution of this problem is the removal of the indefiniteness of the solution.

For Poisson's equation we have $A = B = 1$ and $C = 0$. We set

$$F = \sin \lambda (-3 + 6 \sin^2 \phi)$$

which has a solution of $Q = \sin \lambda \cos^2 \phi$ which is very smooth. Thus we expect this to be a good test of multigrid as the smoother will not start to be effective until we reach the coarsest grids. We compare the number of V-cycles required for convergence and the CPU time taken for this problem for a series of resolutions, each of which has twice as many points in each direction as the previous one. We define convergence to have occurred when the practical smoothing rate is worse than 0.950. The results obtained are in table 3.1.

The column entitled *grids* states how many grids the grid generator produced. For the last example we limited the number to 6 to investigate the effect this would have compared to the 7 grids it would have used. The residuals are calculated as the root mean square of the error on the finest grid as defined earlier. The CPU is the total time taken for all V-cycles whilst the scaled CPU is time taken per point per V-cycle. The code was executed on one processor of a CRAY Y-MP

Dimensions	Grids	Init. Res.	Final Res.	V-cycles	CPU	Scaled CPU
24 x 17	3	.186x10 ¹	.309x10 ⁻⁹	7	0.069	2.42x10 ⁻⁵
48 x 33	4	.247x10 ¹	.173x10 ⁻⁹	8	0.179	1.41x10 ⁻⁵
96 x 65	5	.278x10 ¹	.902x10 ⁻¹⁰	8	0.493	9.88x10 ⁻⁶
192 x 129	6	.294x10 ¹	.438x10 ⁻¹⁰	8	1.532	7.73x10 ⁻⁶
384 x 257	7	.301x10 ¹	.267x10 ⁻¹⁰	8	5.354	6.78x10 ⁻⁶
384 x 257	6	.301x10 ¹	.272x10 ⁻¹⁰	13	8.632	6.73x10 ⁻⁶

Table 3.1: Global multigrid scheme for Poisson's equation.

8-64 using 64 bit precision arithmetic. The maximum possible reduction in the initial residual is 14 orders of magnitude since at this level of reduction we reach the limit of 64 bit machine precision. In practice, because of the restriction, line solvers and associated rounding errors, this is hard to obtain and for this problem we typically get a reduction in the residual of ten orders of magnitude. We note a slight apparent gain in accuracy as the grid size increases; however the reason for this is not obvious and may well be associated with rounding error. Figure 3.3 shows the error for the 384×257 case at convergence with the only visible contour being at 10^{-10} and then only a few points on the equator are visible. Figure 3.4 shows the same picture but with an additional contour at 10^{-11} . We now see a pattern of small scale noise but it is interesting to note that the wave-number 1 pattern of the solution in the λ -direction shows through at this level. We have solved the same problem using 128 bit precision arithmetic and then obtained a solution which converged to over 20 orders of magnitude. This confirms that the main reason behind termination of the iterations is true convergence to machine precision, or as close as is possible.

One property of multigrid is the so called *grid independent convergence*, where increasing the size of the problem does not change the number of cycles required for convergence. This is shown to be true for this case with the exception of the

smallest problem which takes one fewer cycle. Allowing as many grids as possible is obviously beneficial from the example of the 384×257 problem where when we allow one less grid, although the solution still converges to the same accuracy, it takes 5 more cycles. Finally we look at the cost. The cost of multigrid methods usually scales order $n \log n$ where n is the number of points but we see from the scaled CPU that the cost scales better than order n . This is because the CRAY is a vector machine and the bigger problems are more efficient, resulting in increased performance and so comparably cheaper execution. We shall consider vectorisation aspects and the scope for parallelism in more detail in the next section. Without the vector option we see a scaling in line with that expected.

We now consider exactly the same problem but for the hemispheric case where

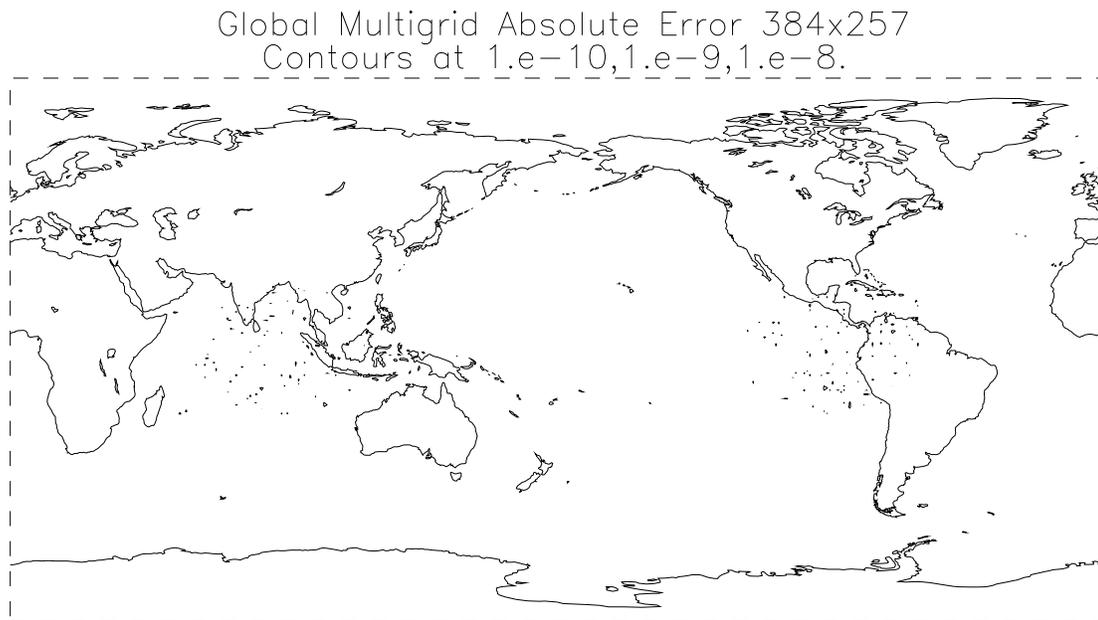


Figure 3.3:

the solution is given at the equator. All the parameters were the same as for the

Global Multigrid Absolute Error 384x257
Contours at 1.e-11,1.e-10,1.e-9.

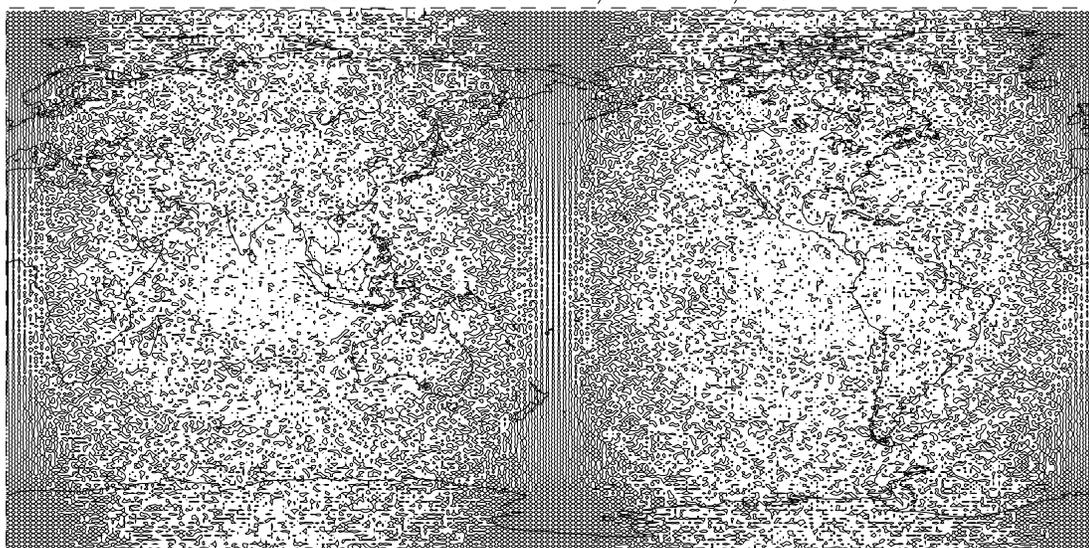


Figure 3.4:

global problem and the results are given in table 3.2.

We note that once again we verify the grid independence assertion for this exam-

Dimensions	Grids	Init. Res.	Final Res.	V-cycles	CPU	Scaled CPU
24 x 17	3	.121x10 ¹	.382x10 ⁻¹³	9	0.113	3.08x10 ⁻⁵
48 x 33	4	.215x10 ¹	.220x10 ⁻¹²	9	0.262	1.84x10 ⁻⁵
96 x 65	5	.263x10 ¹	.151x10 ⁻¹¹	9	0.715	1.27x10 ⁻⁵
192 x 129	6	.286x10 ¹	.941x10 ⁻¹¹	9	2.198	9.86x10 ⁻⁶
384 x 257	7	.287x10 ¹	.541x10 ⁻¹⁰	8	6.877	8.71x10 ⁻⁶

Table 3.2: Hemispheric scheme for Poisson's equation.

ple. The scaled CPU time is greater than for the global problem at each resolution but still decreases as the problem size increases. The reason behind this increase is that we now have shorter vector lengths in the ϕ -direction, in fact half those of the global problem, and the longer the vector the more efficient it is. The unexpected feature is that unlike the global problem the hemispheric problem converges progressively worse as the grid size increases. If we consider the error for the 384×257 problem at convergence, plotted with a contour of 10^{-10} in figure 3.5, we see immediately that the problem is at the equator. Comparing this figure with the equivalent global error, figure 3.3, we note the expected similarity with the exception of this area. The problem therefore lies with the boundary condition and if we contour with an interval of 10^{-3} then we observe that the error is not uniform in sign but has a wave-number 1 signature. In an attempt to remove this we recall that for this problem we used as the boundary condition the analytic solution. However, we should use the actual numerical solution to the problem on the boundary to be consistent, otherwise we have introduced a second order inaccuracy, given that the numerical solution is second order accurate. Doing this, however, does not remove the error. The error appears to be in transferring the problem between the grids and since the solution on any

grid is determined by the boundary condition, the boundary condition must be transferred so as to be consistent with the transferred problem on that grid. The worsening results for the hemispheric problem suggest that more grid transfers equals a poorer solution and that we have not achieved this consistency. For this relatively easy problem it is probably possible to construct a restriction operator which will alleviate this fault. However for the more general problems that we wish to solve, for example equation (3.37) this is not possible and the presence of this error is of some concern.

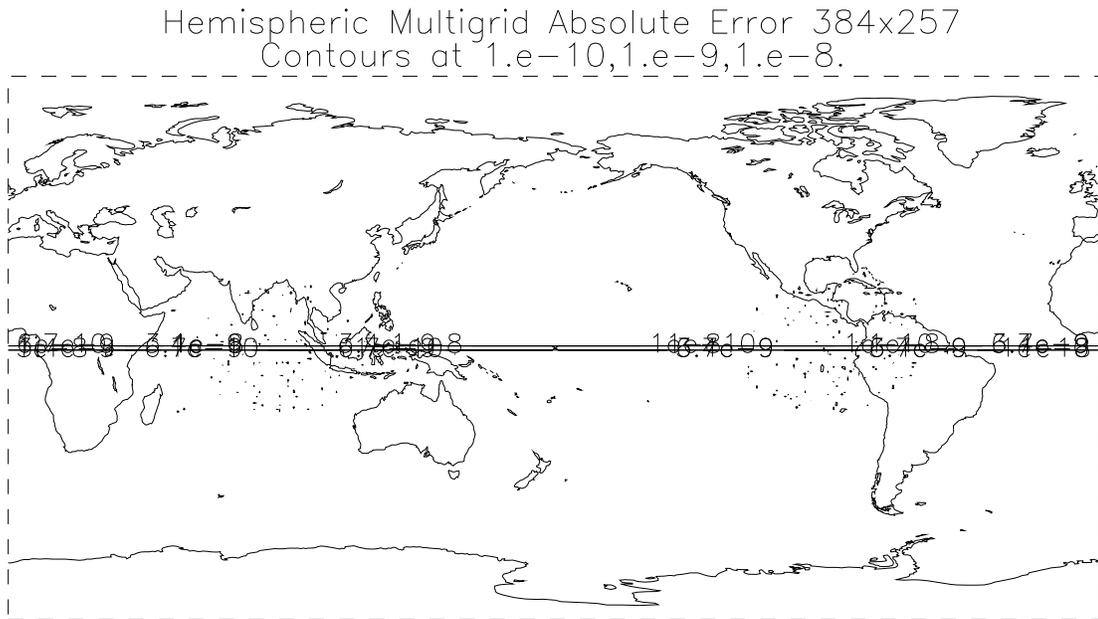


Figure 3.5:

We now consider the more complicated problem where we have variable coefficients

$$A = \frac{2 + \sin \lambda}{\sin^2 \phi}$$

$$B = \frac{1}{\sin^2\phi}$$

$$C = 1$$

$$\begin{aligned} RHS = & -\sin^2\phi\cos^2\phi\sin\lambda + 1. - 2\sin\lambda - 2\sin^2\lambda \\ & + \sin\lambda(4 - 16\cos\phi + 2\cot^2\phi - 4\cos^2\phi\cot^2\phi) \end{aligned}$$

which has as solution

$$Q = \sin^2\phi\cos^2\phi\sin\lambda$$

We note that at the equator the coefficients A and B are infinite as is the right hand side but a smooth analytic solution exists. To find a solution numerically is not possible at the equator since the machine cannot divide infinity by infinity and obtain the correct answer. However we can solve the problem by giving the solution on the equator as a boundary condition. We apply hemispheric multigrid with the analytic solution at the equator to this problem obtaining the results in table 3.3.

These results are very similar to those gained with Poisson's equation and ex-

Dimensions	Grids	Init. Res.	Final Res.	V-cycles
24 x 17	3	.857x10 ¹	.254x10 ⁻¹²	8
48 x 33	4	.193x10 ²	.119x10 ⁻¹¹	10
96 x 65	5	.448x10 ²	.553x10 ⁻¹¹	10
192 x 129	6	.117x10 ³	.251x10 ⁻¹⁰	10
384 x 257	7	.325x10 ³	.113x10 ⁻⁹	11

Table 3.3: Hemispheric scheme for variable coefficient problem.

hibit the same failing in that there is a large error at the equator compared to elsewhere in the domain. However, the hemispheric code does produce a solution whereas the global problem is inapplicable. The problem could be re-scaled to allow a global solution by multiplying by the reciprocal of the largest coefficient

throughout, provided that this is not zero except possibly at the equator. However, for general coefficients, there is the problem that they are inside a partial derivative and care must be exercised in this re-scaling to avoid leaving divisions by zero in the code. In practice the coefficients need to be differentiated out to leave a system which includes terms involving both first and second derivatives of the solution and first derivatives of the coefficients. To apply centred differences to this system requires averaging of derivatives and coefficients in parts of the discretisation which can lead to a loss of accuracy.

We now return to the question of whether the (2,3,2) V-cycle is optimum and also whether or not we require alternating line relaxation. To investigate these questions we consider again Poisson's equation on the sphere using the hemispheric version of multigrid at 384×257 resolution. Due to changes in the maths library routines used to calculate trigonometric functions on the Cray Y-MP the result for the (2,3,2) V-cycle with alternating line relaxation is not identical to the one obtained earlier. All comparison results quoted here and elsewhere were performed consistently, in the sense that any results compared used the same maths library routines. For the results in table 3.4 all the schemes converged to the same degree of accuracy.

Alternating (A) or λ -line (λ)	V-Cycle	Number of Cycles	CPU time in seconds
A	(2,3,2)	9	7.856
λ	(2,3,2)	9	3.542
λ	(1,1,1)	16	3.630
λ	(1,3,1)	15	3.311
λ	(1,3,2)	10	3.029

Table 3.4: Comparison of Multigrid schemes.

From table 3.4 it is easy to see that for this problem there is no need for the alternating line scheme as the λ -direction line solver on its own converges to the same accuracy in the same number of cycles. Using the λ -direction only solver reduces the cost by over 50% in this example. To test if this held true for solving the correction equation (3.37) from the previous chapter idealised data runs were performed with the alternating line and λ -direction only schemes using the (2,3,3) V-cycle. The following times were obtained for the first order in time semi-Lagrangian solution of the McDonald-Bates idealised problem, described in the next chapter, with a 1800 second timestep and run for 5 days. The times are the total times for the run and so include all code.

Alternating lines : 1675 seconds

λ -direction only : 941 seconds

We again see a significant saving, 44% in the execution time by using the λ -direction only solver. The difference in the height solutions at the end of the integration, caused by the slightly different results obtained by the two different multigrid procedures, is of the order 0.001 metres which is negligible.

We now look again at table 3.4 to see the effect of changing the V-cycle properties on the λ -direction only scheme. The (1,1,1) V-cycle and the (2,3,2) V-cycle take almost identical times with the (1,1,1) cycle taking approximately twice the number of cycles to converge. From this we can deduce that it is not the grid transfers that matter in this problem but the number of actual solves the scheme performs, since the total number of solves performed by each version was almost the same. Since the problem is known to be very smooth we try the (1,3,1) cycle to see if there is any improvement from solving more accurately on the coarsest grid. It is not surprising therefore that this scheme is faster than the (1,1,1) scheme but only by 8%. Finally we try the (1,3,2) scheme since in many practical implementations of multigrid the optimal computational scheme has been found to be one where extra solves were included on the ascending branch while keeping

to just one on the descending branch. This scheme produces a further saving and for this problem the (1,3,2) V-cycle λ -direction only scheme has been found to be optimal on one processor of a CRAY Y-MP. Using the (1,3,2) V-cycle λ -direction only scheme for the idealised McDonald-Bates problem gives a CPU time of 925 seconds which is a modest saving compared to that gained by not using the alternating line scheme. Again no significant impact on the solution was noticed with this version of multigrid and we can use any of the multigrid schemes to obtain the solution if we are not concerned about the amount of CPU time used.

3.3.5 Discussion of future developments and parallelisation aspects.

We intend to extend the code to 3-dimensions but before proceeding too far in this direction it is necessary to consider performance aspects closely. For the moment we shall be running the code on a CRAY Y-MP 8-64 computer which has 8 processors and 64 megawords of memory. The 2-D code used only 1 processor and only just over 3 megawords of memory for the largest problem we considered in the previous section. We shall ignore memory considerations as these are not of primary concern. The question is how to make most use of the computer architecture to obtain the best execution time. The CRAY uses a shared memory architecture and when using microtasking it behaves as an SM-SIMD machine. Shared Memory Single Instruction Multiple Data, hence SM-SIMD, is based on the principle that all memory is central and shared, with the same piece of work being allocated to each processor but working on a different piece of data. The best example is a loop construct. Suppose we have a loop over 800 elements and we simply wish to perform

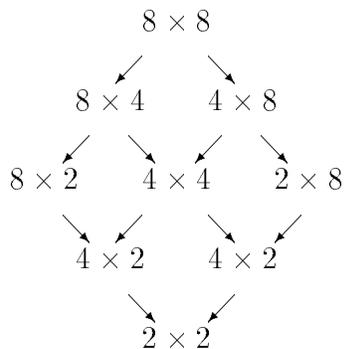
$$A = B + C$$

The machine allocates the first hundred elements to processor one, the next hundred to processor 2, and so on. They then each perform the calculation in parallel on their own area of data and then return the answer to main memory. In simple terms we can gain parallel performance if the calculations we want to perform do not feed back on each other. We note that if a loop will run in vector mode then it will also execute in parallel. To implement multigrid on a SM-SIMD machine and obtain good performance is relatively easy if we use a red/black or zebra type scheme. These have the desired advantage of not interacting. Performance however becomes less efficient as we coarsen the problem as there is less work to spread around. For the 2-D cases considered in the last section there was no noticeable benefit from using more than one processor. In fact for the small problems the cost actually increased since to perform parallel execution incurs some cost penalties. In 3-D this will be much less of a problem mainly due to the increased problem size.

Another form of parallel architecture is known as DM-MIMD, Distributed Memory Multiple Instruction Multiple Data. Here we have a connected network of processors with no central memory which each have the entire instruction set but only a small amount of the data. The main problem with making multigrid run efficiently on this architecture is that we come across the *granularity problem*. To make any code efficient the amount of computation done by each processor must be significantly more than the time spent communicating with its neighbours. Communication is considerably more expensive than computation and the number of times information is required about the new values of points surrounding any processor thus needs to be minimised. The granularity is the number of nodes of the problem held by each processor to enable this computation versus communication battle to be won. This suggests that point smoothers are the desired option and that line smoothers, where we need to solve a large matrix equation are likely to be impractical if communication overheads are very large.

Regardless of the smoother we use as we coarsen the grids we leave fewer and fewer points at each node and performance deteriorates. One way to get around this is to limit the minimum number of points each processor can have, however this restricts the number of grids we can generate and we have already seen how this can significantly increase the number of iterations required. To make things even worse as we add more processors to increase our computing power the performance of multigrid can get worse. One solution is to use a system called *agglomeration* where we keep the granularity high by coarsening and keeping the numbers of points on some processors large at the expense of leaving some with no points at all. Thus we are effectively reducing the number of processors working. This has the disadvantage, in practice, of increasing the length of the communication links but can gain some extra efficiency and does not increase the number of iterations required. Smith and Fiddes [71] noted that implementing multigrid without agglomeration was significantly more complicated than other iterative solvers, for example conjugate gradient, on a DM-MIMD machine. Other authors, for example Linden et al. [40] and Robinson [62], have successfully implemented multigrid on parallel architectures with varying degrees of success and much work will no doubt continue in this area. However, of more concern is that no one has considered, as far as we know, the problem of including line solvers rather than simple point solvers. Naik and Van Rosendale [55] have suggested that a technique called *semi-coarsening* can be used in conjunction with point solvers to avoid the need for line solvers. Semi-coarsening advocates coarsening only in directions in which the smoother performs well. They have successfully implemented this approach for problems with strong anisotropy. However Barros [6] mentions this idea but concludes that in cases where the anisotropy changes direction inside the domain, as it does for our case, semi-coarsening may be difficult and expensive to implement. This is because of the changing anisotropy causing different strategies to be required at different points in the mesh. In the worst

case semi-coarsening degenerates to the so-called *algebraic* multigrid method. In this method we generate all possible grids by restricting all current ones in one direction at a time. We then solve on each. This requires a more complicated set of restriction and prolongation operators as we need ways to combine answers from various grids. A simple 8×8 example of all the grids generated, assuming a minimum number of 2 points in each direction, exhibits all the salient features;



We note for example that to obtain the problem on the 4×4 grid we need to combine the information from the 2 grids above it. This combination is not necessarily a simple addition but depends again on anisotropy and the problem. However, it may prove to be worth the effort on a DM-MIMD machine where line solvers will probably be impractical.

Chapter 4

Computational Results

In this chapter we present computational results for several idealised and real data cases using the predictor/corrector method described in the last chapter. We compare the performance of the various advection schemes on one of the idealised problems, and also on real data. We also compare results with those obtained from a C-grid primitive equation model, the semi-Lagrangian model of Bates et al. [7], henceforth PE. Before we do this we need to explain how initial data is obtained for both models, as initialising the SSG model, particularly with real data, requires some care.

4.1 Data Initialisation

A. SSG Model.

The shallow-water SSG model requires consistent initial fields of h , \underline{u}_0 and \underline{u} . Given an initial height field, providing it is inertially stable and q is not zero, see section 2.2.2 for the definition of q , then we could obtain all the other fields directly from the equations after having calculated the initial values of $\partial h/\partial t$

and hence $\partial \underline{u}_0 / \partial t$ by solving the elliptic equation for $\partial h / \partial t$. However the initial height field may not satisfy these constraints, as typically happens when using 500 hpa height fields from operational forecast models. Also many idealised problems used to test the shallow-water equations suffer from the same failing, for example the problem of McDonald and Bates described in section 4.2.1 is inertially unstable near the equator, and the Rossby-Haurwitz wave problem, section 4.2.2, has areas of negative q . We cannot use the direct method to find the other fields when we do not have an initial height field which satisfies the constraints, and so we have developed the following initialisation procedure for the SSG model.

Initialisation stage 1.

We take the initial height field and iterate the correction step from the predictor/corrector method but updating only the balanced velocity field \underline{u}_0 . We perform k iterations of this procedure which gradually fits \underline{u}_0 to the height field where possible. In areas where the height field violates the conditions required to find the other fields directly, the inertial stability constraint applied to the \underline{u}_0 field prevents the balanced wind field from satisfying the balance equations for this illegal height field. The procedure can be summarised as

iterate k times

Step 1 Corrector step. Updating \underline{u}_0 only.

Step 2 Inertial stability enforced.

Initialisation stage 2.

The second stage of the procedure is to take the fields at the end of the first stage and apply the correction step procedure again but now allowing h to be updated

as well. In areas where a valid solution could have been found by the direct method, the first stage of the initialisation should have also found that solution provided that k was large enough. In areas where a valid solution could not have been found a residual still exists and the second stage of the initialisation removes this residual without significantly affecting the solution in the other areas. The procedure is thus

iterate l times

Step 1 Corrector step. Updating \underline{u}_0 and h only.

Step 2 Inertial stability enforced.

We now need define the initial values of \underline{u} which we set to be equal to the balanced velocity obtained at the end of the second stage of the initialisation. In principle we should calculate $\partial h/\partial t$ and $\partial \underline{u}_0/\partial t$ and find \underline{u} exactly from the direct method, but in practice setting the initial values equal to the balanced velocity has been found to be sufficient.

The only serious problem with this method was found when trying to initialise the model with a uniform zonal flow. In this case the procedure successfully initialised the flow in mid to high latitudes but was very poor at initialising in low latitudes. This is because the rate of convergence of the initialisation procedure when h updates are not allowed is proportional to f , and hence the number of iterations needed to obtain the solution is proportional to f^{-1} and near the equator tends to infinity. This problem is easily overcome by setting the uniform zonal flow directly since the height field satisfies the conditions required to use the direct method. In problems, such as the Rossby-Haurwitz problem, where a perturbation is applied to a uniform zonal flow we use the initialisation procedure after first setting the balanced velocities to satisfy the uniform zonal flow part of

the initial data.

For all the problems we consider in this section typical numbers used in the initialisation procedure are, 60 iterations of the first stage of the initialisation procedure and 30 iterations of the second stage. The most important thing to achieve is to fit the balanced wind field to the height wherever possible and experiments show this to be well achieved after 60 iterations. Only a few iterations of the second stage are usually required, typically 5 to 10, but we use a lot more than is required simply to be sure that the initial residuals are removed to machine precision.

B. PE Model.

Primitive equation models are susceptible to gravity waves which are a solution not supported by the SSG model. For many problems gravity waves can be generated by imbalances in the initial data, and appear as noise in the solution, which can lead to difficulty in interpreting the underlying solution to the problem. Two approaches are frequently used to overcome this problem. The first is to add a damping operator to the equations to suppress the gravity wave noise. This has the disadvantage of also introducing some unwanted smoothing into the underlying solution as well. The second approach is to use an initialisation scheme to remove the initial imbalances. This second approach is generally successful for some time period into the forecast, after which gravity wave noise can appear because of imbalances generated by forcing functions. For the unforced shallow-water equations there are very few imbalances generated by the numerical procedure, and a successful application of an initialisation procedure is usually sufficient to suppress the gravity waves for the whole forecast period in which we are interested, typically upto 10 days ahead. We use the digital filter initialisation

procedure of Lynch and Huang [41] to initialise the PE model. For the idealised data cases, where the initial wind fields are basically geostrophic, the initialisation procedure has little impact and both PE and SSG models run from very similar initial data. For the real 500 hpa data the different initialisation procedures can lead to some significant differences between the two starting states. This will be an issue we shall address when we consider the real data cases in section 4.3.

4.2 Idealised Data

4.2.1 McDonald-Bates 1989 Problem

The problem we consider was posed in McDonald and Bates [44] and is defined by setting the initial height field h by

$$h(\lambda, \phi) = \bar{h} + \frac{af\bar{v}\sin^2\phi\sin\lambda}{g}$$

where \bar{h} and \bar{v} are fixed constants taken as $\bar{h} = 57680/g$ metres and $\bar{v} = 20ms^{-1}$, with f the Coriolis parameter and g the acceleration due to gravity. This data represents a high/low system in each hemisphere which forces a flow of \bar{v} metres per second across the pole. The balanced wind field \underline{u}_0 and the full wind field \underline{u} were then set via the procedure described in section 4.1 with 60 iterations used in both stages of the initialisation procedure. If we set the initial velocities directly using the balance equations we find that the data is inertially unstable across the equator. This can be verified by substituting the analytic form for h into the inertial stability conditions. The initialisation procedure removes this instability whilst having negligible impact on the initial data elsewhere on the sphere. Figure 4.1 shows the initial height field obtained in the northern hemisphere. Since the data is identical in the southern hemisphere, except that the high/low pair are rotated through 180 degrees of longitude, the solutions are identical in each hemisphere so we show charts only for the northern one. We now integrate the ini-

tial data using the predictor/corrector method of the previous chapter with three different prediction schemes. The first is the semi-Lagrangian scheme with first order accurate trajectories; we will consider the effect of second order accurate trajectories in the section on real data problems. The second is the Heun scheme with Fourier damping and the third is the Heun local timestepping scheme. All three schemes were run at a spatial resolution of 192×129 points with a 1800 second timestep and output was obtained after 5 days of integration. Figure 4.3 shows the height field after 5 days for the semi-Lagrangian run. Comparing this with the initial height field, figure 4.1, we note that the high has gained height whilst the low is less intense. This is unexpected since we assumed that the high and low would both become weaker by a similar amount, since this is what was found by McDonald and Bates [44] running their primitive equation model. We will verify their results later in this section using the same resolution and timestep used here. One property of the SSG equations on the sphere seems to be their ability to maintain high pressure systems at their initial intensity for a long time; we shall see this again when we look at the real data problems. The same results are obtained in the height fields from the other prediction schemes, figures 4.5 and 4.7, with a summary of the values in table 4.1. We will consider this behaviour more fully later, when we compare results with those for the primitive equations. For the moment we consider only the differences between the different integration schemes used here.

From the height fields after 5 days it is hard to see any differences between the three schemes, so we look at a field which is more sensitive to any small scale noise. We consider the SSG absolute vorticity, q , see section 2.2.2 for definition, which is a quantity obtained from differentiating the fields and shows greater sensitivity to any noise. q is also an important field to look at since the existence of solutions depends on it satisfying the ellipticity condition, equation (2.22), and

Scheme	Max Height (m)	Min Height (m)
Initial Data	6494.76	5264.83
Semi-Lagrangian after 5 days	6517.20	5382.72
Heun Fourier Damped after 5 days	6517.19	5382.83
Heun Local Timestepping after 5 days	6517.26	5382.72

Table 4.1:

any scheme prone to noise is likely to come close to violating this condition more often than one that is not. The initial q field is shown in figure 4.2, and we see that it is very smooth with approximately circular contours with a maximum value of 39.07×10^{-13} . Figure 4.4 shows the same field after 5 days integration using the semi-Lagrangian scheme. We see that the contours have remained very smooth and retained the same shape, with the exception that there is some ridging in the contours under where the high in the height field can be found. The maximum value has also increased slightly to 40.34×10^{-13} . Figure 4.6 shows the same field for the Fourier damped Heun scheme, and at first sight there appears to be little difference between it and the semi-Lagrangian solution. On careful inspection it can be seen that the 40 unit contour around the maximum value is not as smooth and circular as was the case with the semi-Lagrangian solution, showing that there is some degradation of the solution very close to the pole. Figure 4.8 shows q for the local timestepping Heun scheme. Here we can see significant noise extending as far as 55 degrees North, and becoming particularly bad near the pole where there are values in excess of 140×10^{-13} . This noise was not particularly sensitive to the number of local timesteps used and increasing the number did not remove the problem. We do not consider the local timestepping option further because of this noise problem, but to help choose between the other two schemes we re-run the same problem using a timestep of 3600 seconds. Again the height

fields, figure 4.9 for the semi-Lagrangian scheme and figure 4.11 for the Fourier damped Heun, look very similar. Comparing them with those obtained for the 1800 second timestep run we notice a further increase in the height at the centre of the high to 6534 metres, the low is also more intense. Looking at the q fields, figures 4.10 and 4.12, we notice that away from the pole they again agree very well, but whilst the semi-Lagrangian solution still has a smooth circular 40 unit contour the Fourier damped Heun scheme has started to fragment this feature.

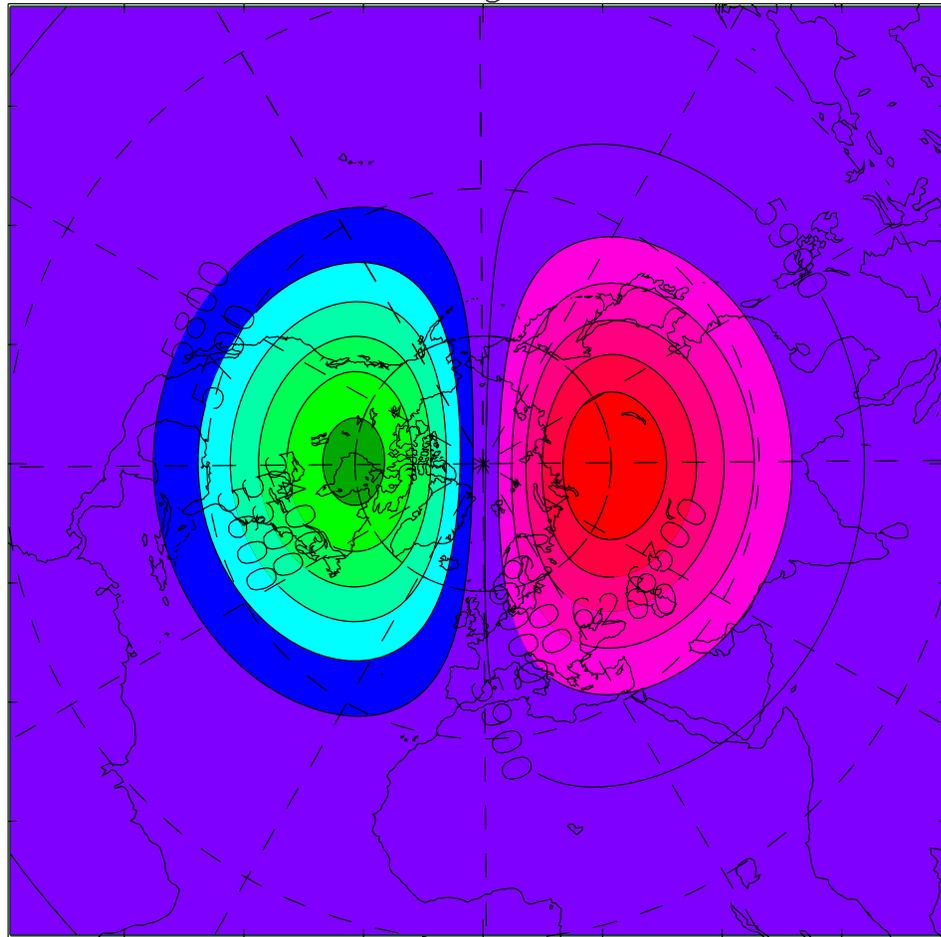
To further compare the various schemes we investigate the accuracy of the schemes for real data. Here we use the real data simply to look at accuracy and leave the discussion of the results until section 4.3. 2 day integrations from the initial data of 1st February 1991 were performed, running with a 1800 second timestep and we take as the true solution a semi-Lagrangian run with a 300 second timestep and second order accurate trajectories. We evaluate the mean and max height differences between the three schemes we have looked at so far, plus the semi-Lagrangian scheme with second order accurate trajectories, and the calculated truth at the end of the 2 day integration. The results are in table 4.2.

Scheme	Mean Height Error (m)	Max Height Error (m)
S-L 1st	2.014	31.205
S-L 2nd	1.952	31.670
Heun FD	2.647	35.252
Heun LT	2.800	38.371

Table 4.2: Comparison of advection scheme errors.

We see that the semi-Lagrangian schemes have a considerably lower error than the Eulerian ones, although because the numbers are relatively small it is hard

QE Model. MB89 Data
Initial Height Field.



5300 5500 5700 5900 6100 6300 6500

Figure 4.1:

QE Model. MB89 Data
Initial q Field.

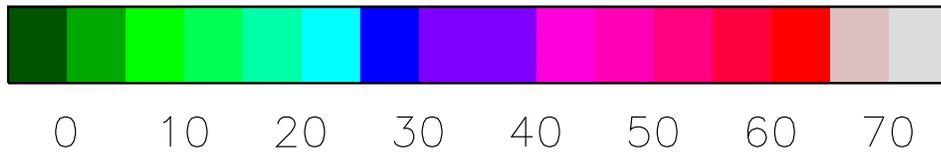
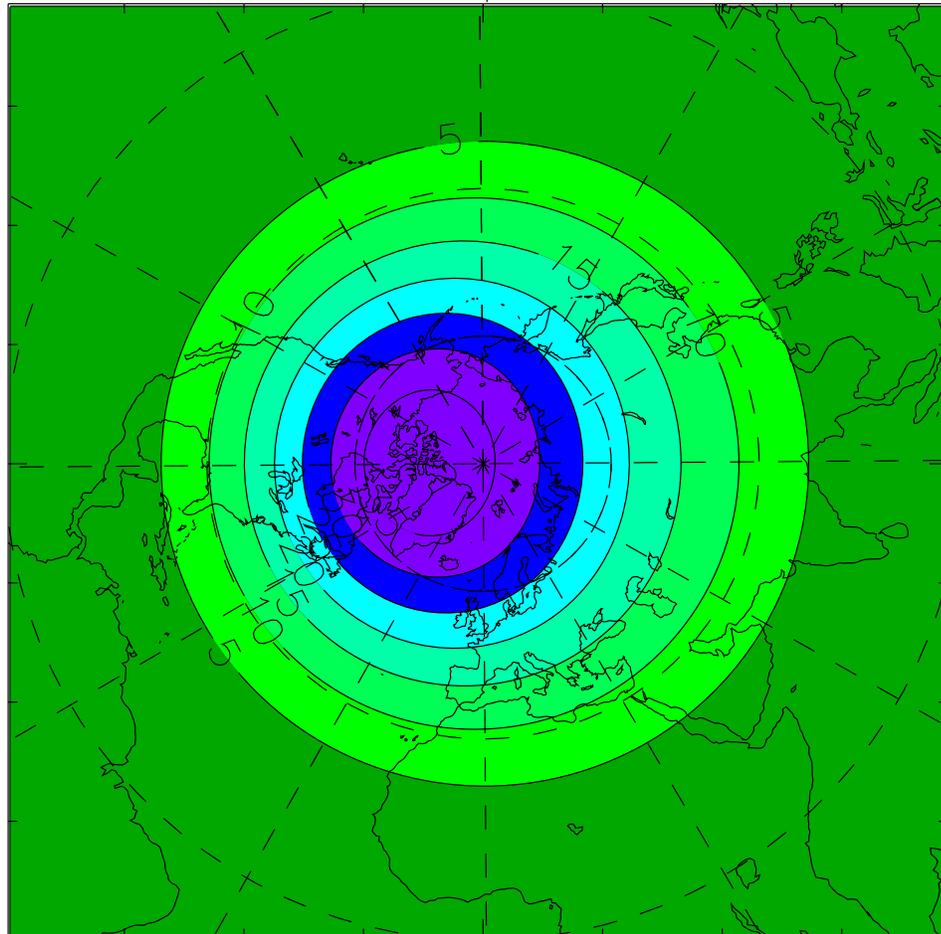
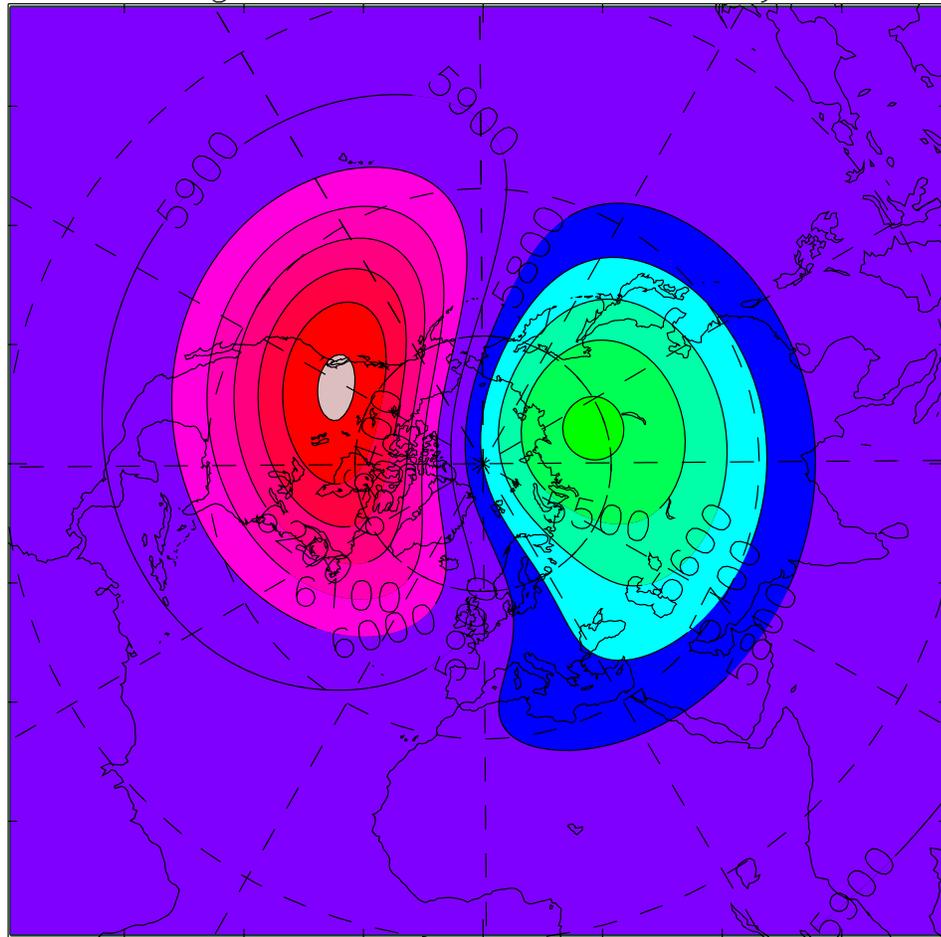


Figure 4.2: Contour interval = 5×10^{-13} .

QE Model. Semi-Lagrangian. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.3: Timestep=1800 seconds.

QE Model. Semi-Lagrangian. MB89 Data
q Field after 5 days.

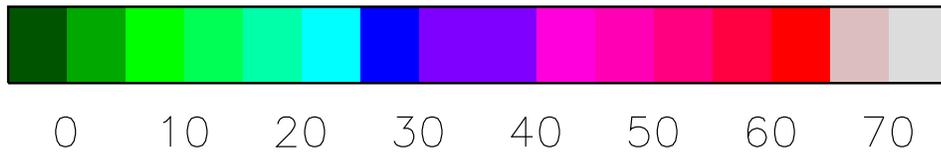
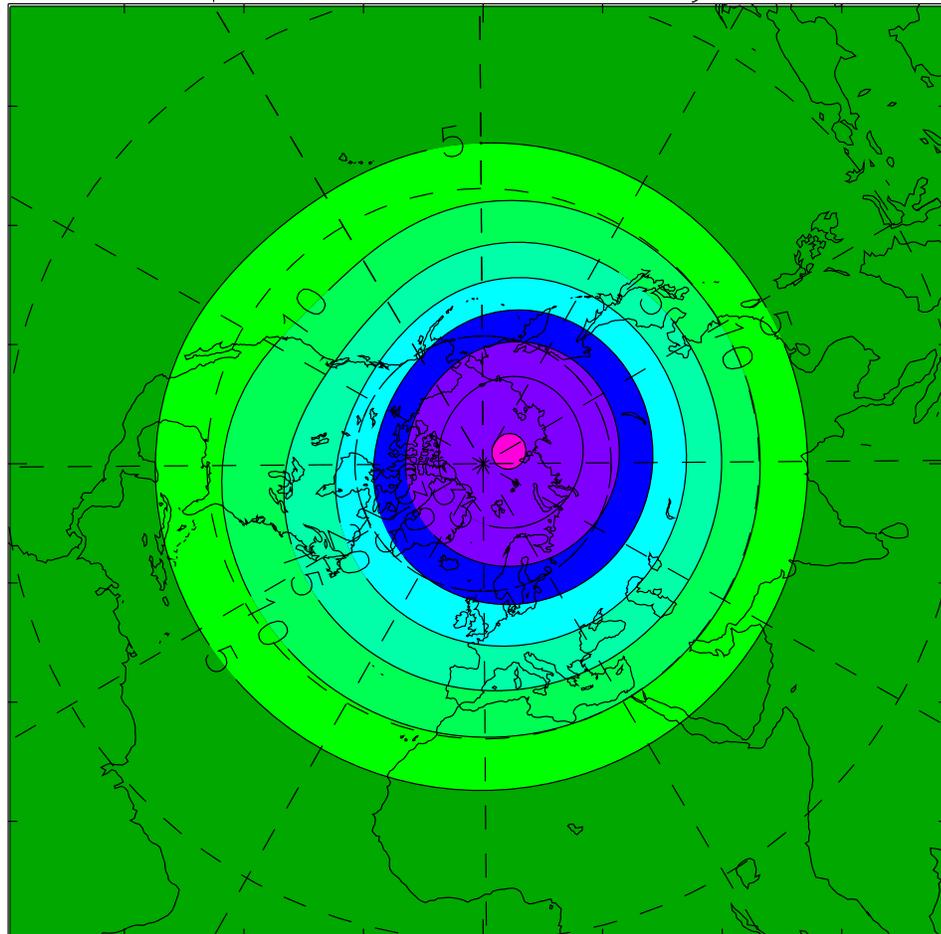
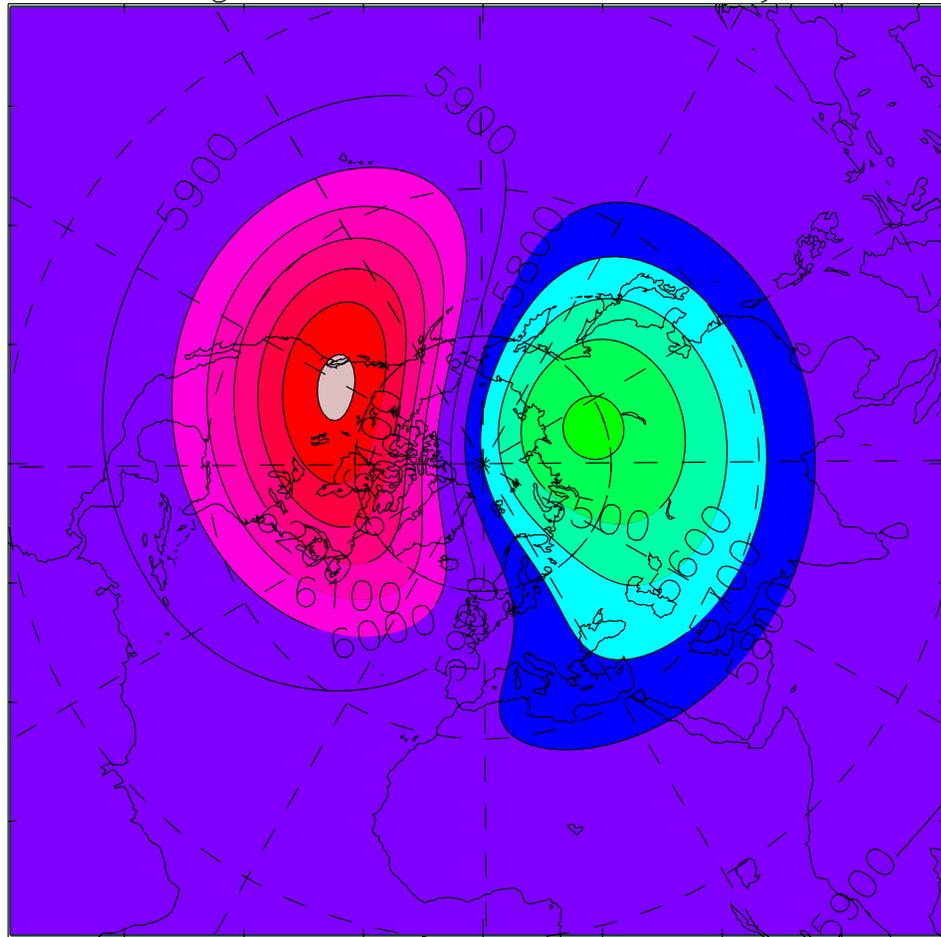


Figure 4.4: Timestep=1800 seconds. Contour interval = 5×10^{-13} .

QE Model. Fourier Damped Heun. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.5: Timestep=1800 seconds.

QE Model. Fourier Damped Heun. MB89 Data
q Field after 5 days.

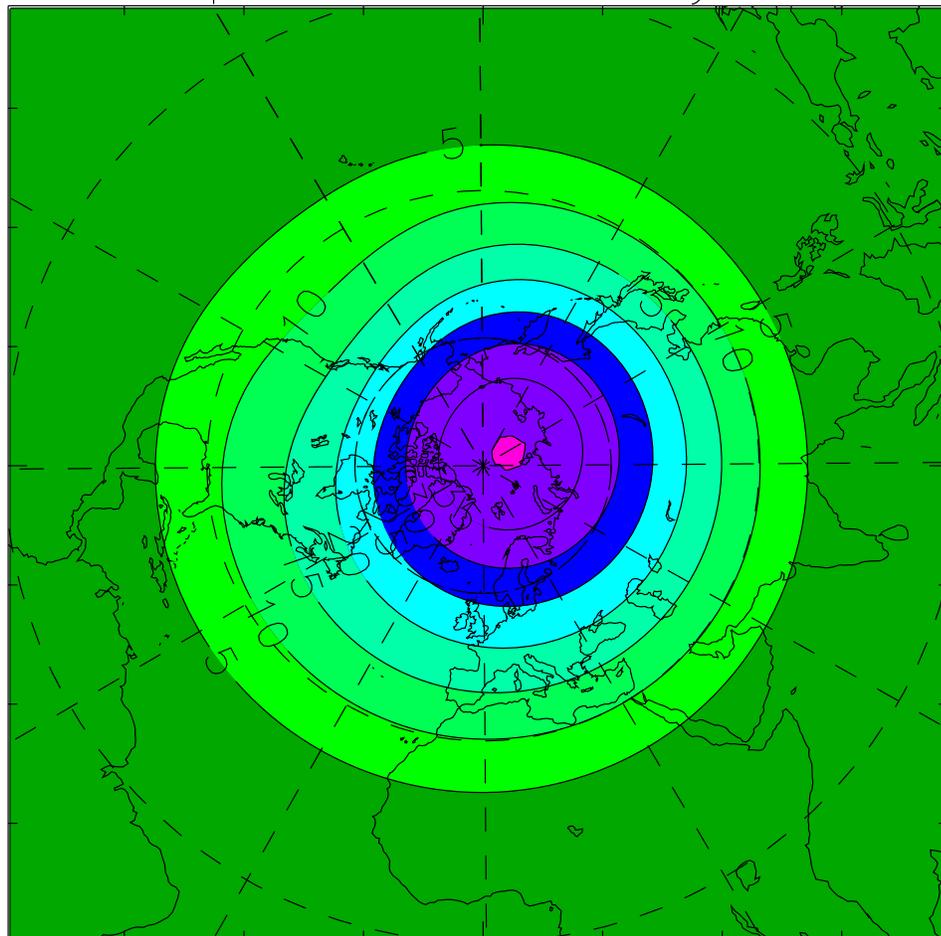
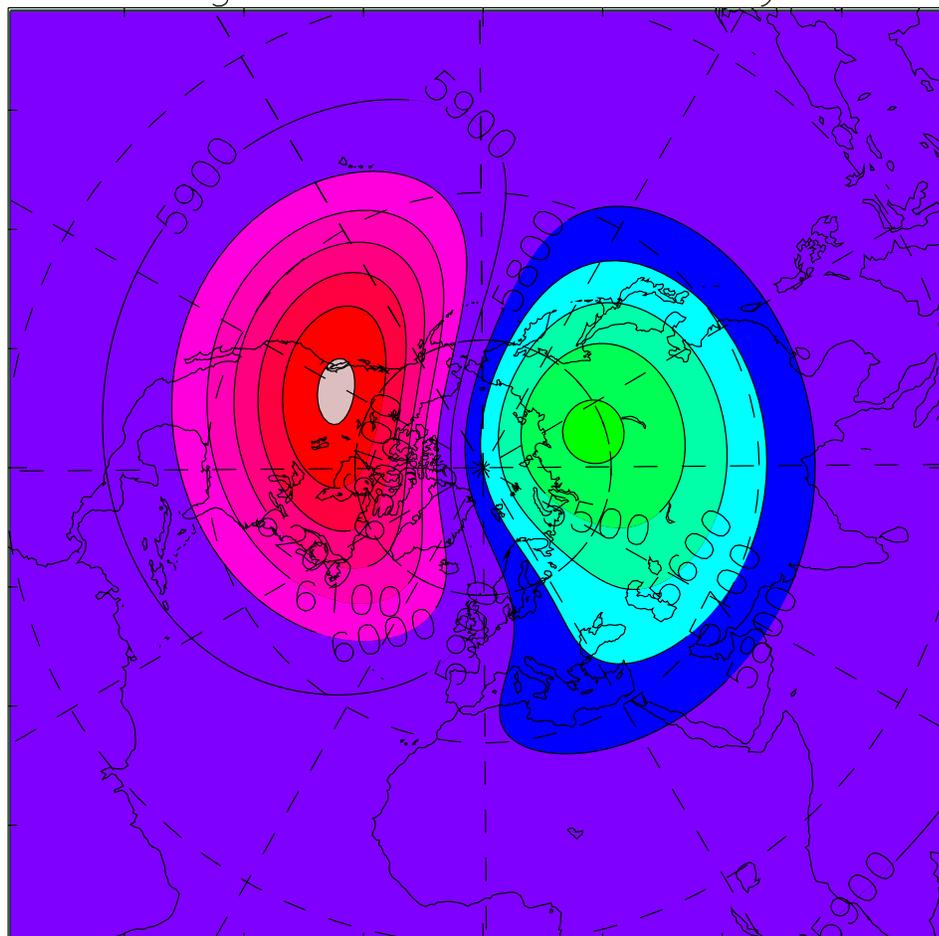


Figure 4.6: Timestep=1800 seconds. Contour interval = 5×10^{-13} .

QE Model. Local Timestepping Heun. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.7: Timestep=1800 seconds.

QE Model. Local Timestepping Heun. MB89 Data
q Field after 5 days.

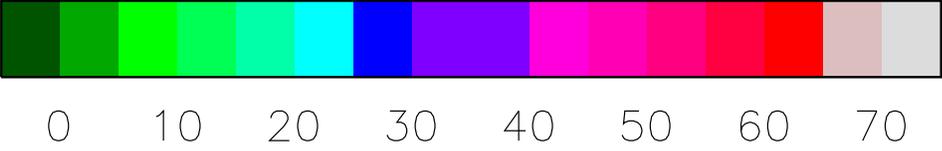
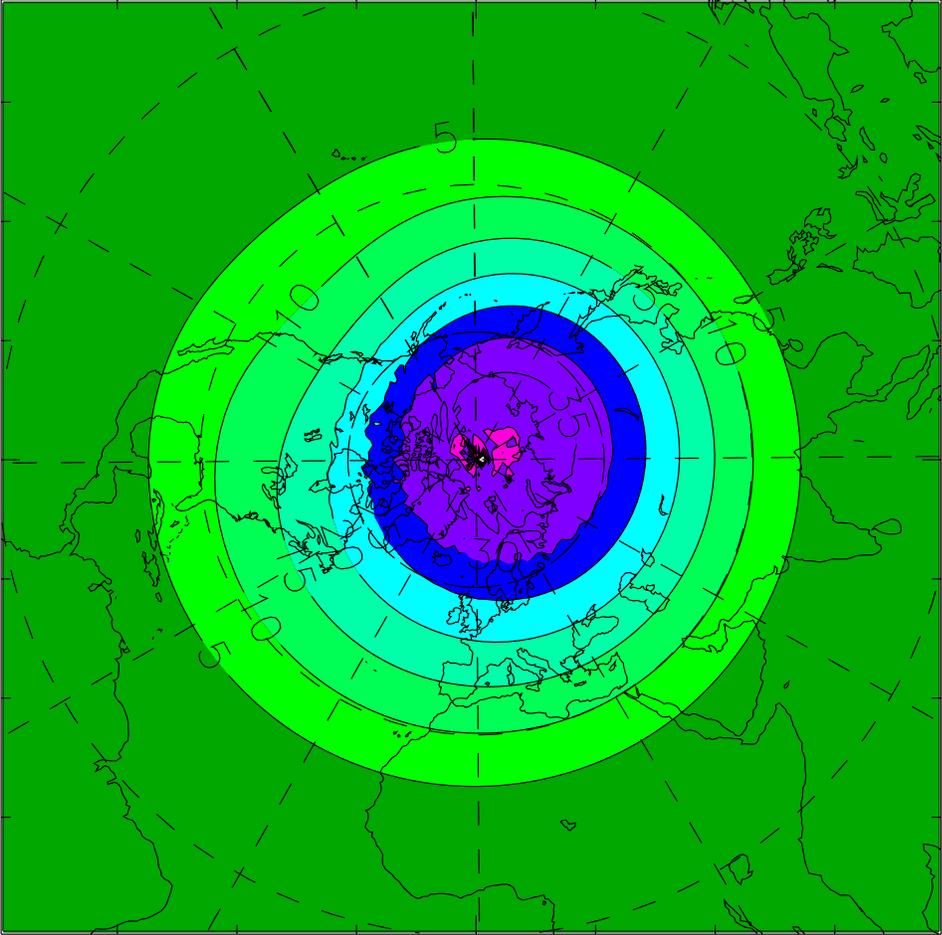
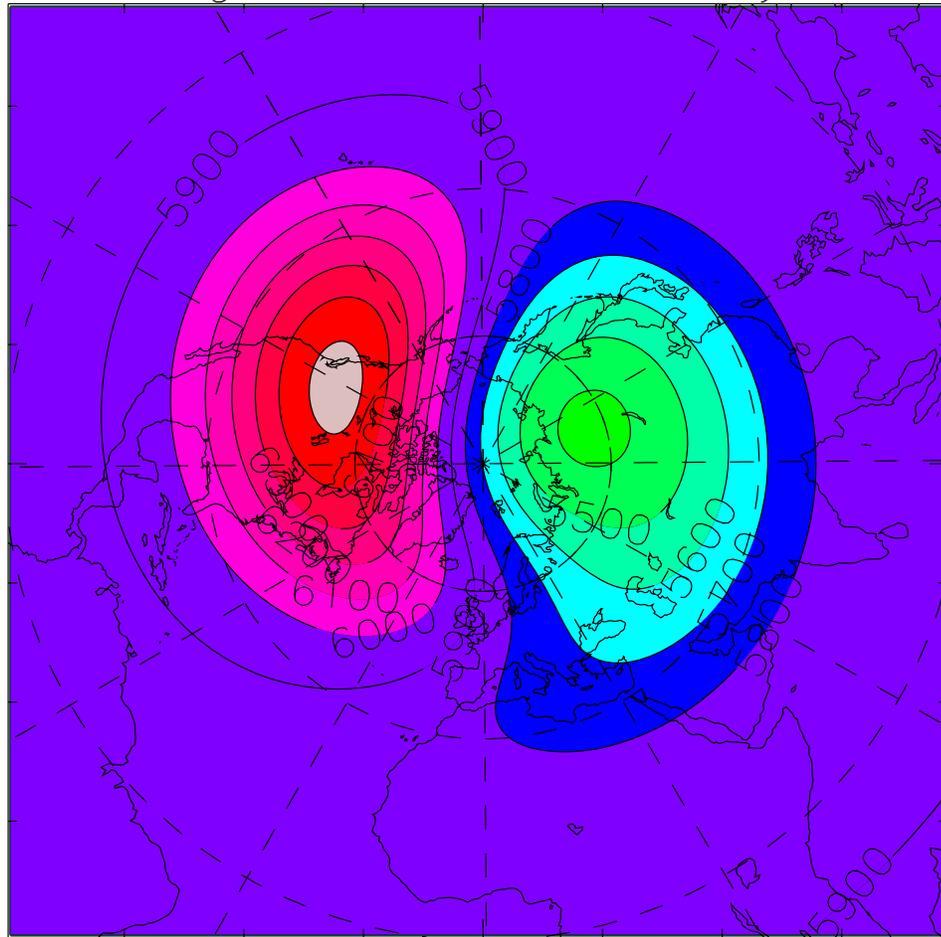


Figure 4.8: Timestep=1800 seconds. Contour interval = 5×10^{-13} .

QE Model. Semi-Lagrangian. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.9: Timestep=3600 seconds.

QE Model. Semi-Lagrangian. MB89 Data
q Field after 5 days.

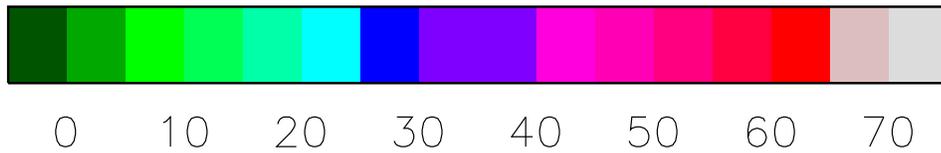
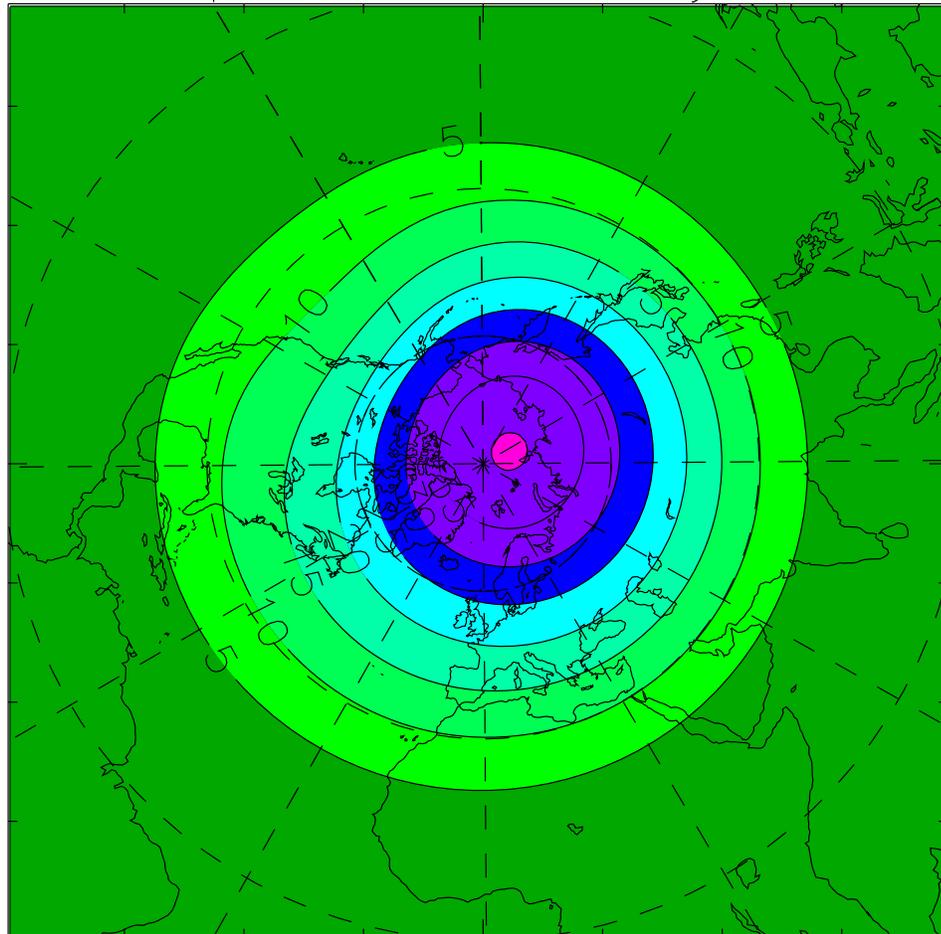
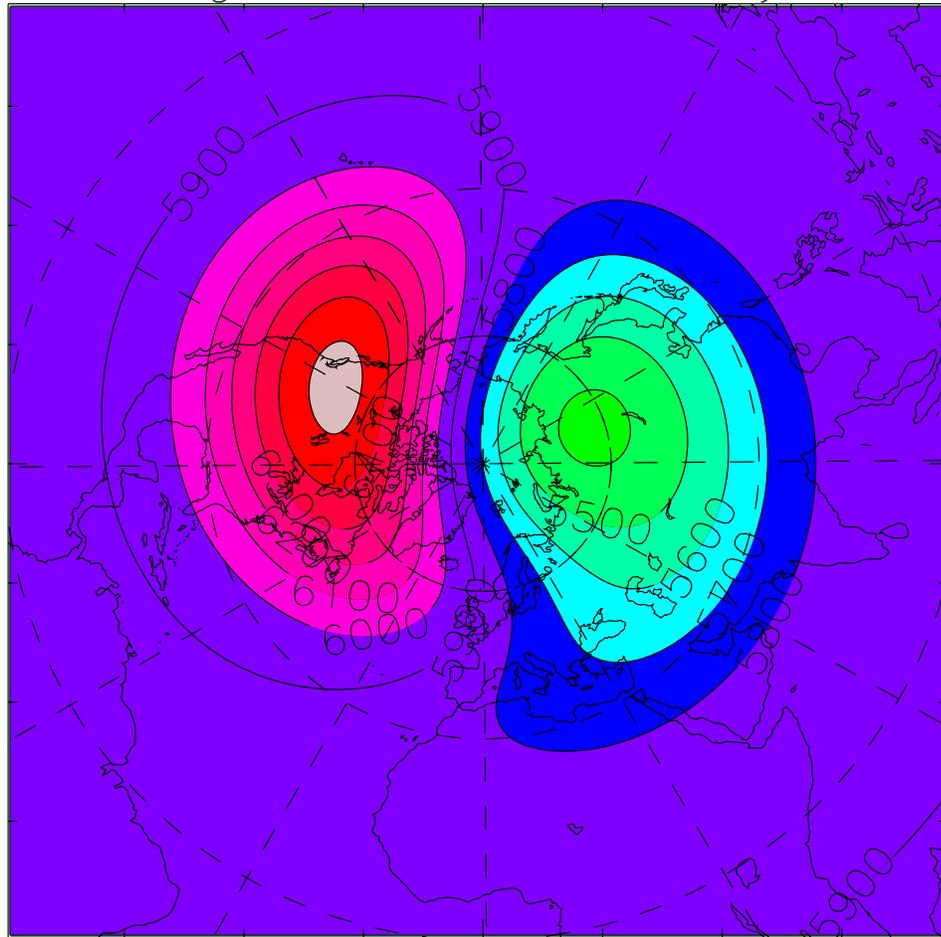


Figure 4.10: Timestep=3600 seconds. Contour interval = 5×10^{-13} .

QE Model. Fourier Damped Heun. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.11: Timestep=3600 seconds.

QE Model. Fourier Damped Heun. MB89 Data
q Field after 5 days.

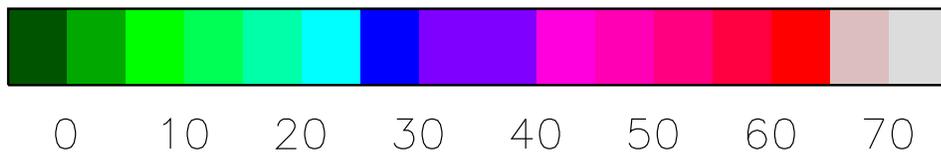
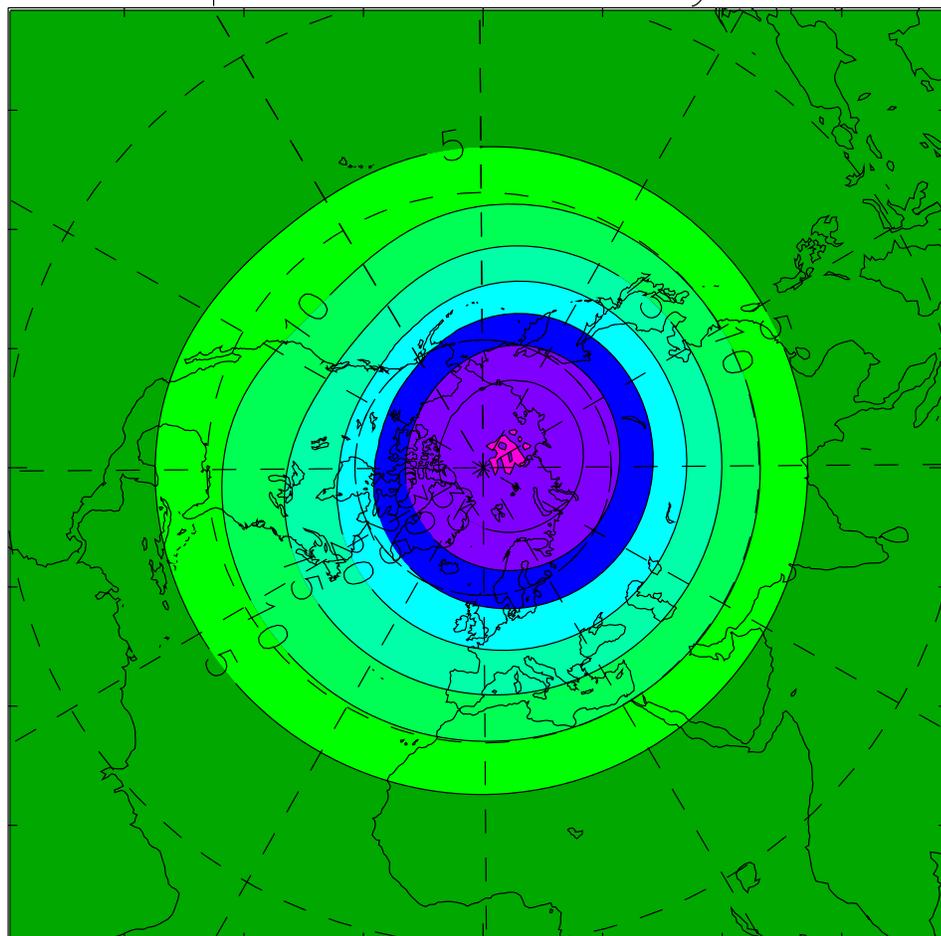


Figure 4.12: Timestep=3600 seconds. Contour interval = 5×10^{-13} .

to see the differences when looking at charts of h , except in very small areas. The difference between the two semi-Lagrangian schemes is less significant, but the second-order accurate trajectories do yield a benefit for this timestep which agrees with the findings of McDonald and Bates [43]. We choose to use the semi-Lagrangian method from now on since we have seen that it is better able to represent features in the q field near the poles and has smaller errors. This has an added advantage in that when we compare with the primitive equation solutions we shall be using the same integration method in both models.

We now consider how the McDonald-Bates solutions from the SSG model compare with those from a primitive equation model. The PE model we use is the C-grid shallow-water semi-Lagrangian model of Bates et al. [7] upon which the semi-Lagrangian SSG model was based. The only major difference in the implementation of the semi-Lagrangian scheme in the PE model is that we use the semi-Lagrangian form of the mass equation, and not the Eulerian form used in the SSG model. This means that the PE model does not explicitly conserve mass, although deviations from the initial global mean value are very small. Figure 4.13 shows the height field after 5 days from the PE model with an 1800 second timestep. The initial field is almost exactly the same as for the SSG model, figure 4.1, and so we compare the solution with the initial data from the SSG model. We note that both high and low have lost intensity almost equally, approximately 110 metres. These solutions verify well with those found in Bates et al. [7]. Comparing the PE solution with the corresponding SSG semi-Lagrangian solution, figure 4.3, we see that unlike the PE model the high in the SSG model has grown slightly whilst the low has lost almost exactly the same intensity, 118 metres. The positions of the centres of the high and low agree quite well with the high very close to the same location, whilst the low in the SSG model is approximately 7 degrees east of the PE centre. Since the solution is rotating clockwise this implies

that the low in the SSG model is rotating at approximately 1 degree per day slower than the PE low. The low in the SSG model is more circular than the PE solution which shows a south-west to north-east slant. Apart from this, the shape of the features is broadly similar but with the features covering a slightly different area. This is because the SSG model has to maintain mass conservation, and since it has not lost the central mass in the height field to balance the gain in the low, it must change the areas of the features to ensure conservation. To see why there are differences in the high in the two models we would like to look again at the SSG vorticity, q/f . q/f is defined in terms of the balanced velocities which are not available from the PE model. Instead we shall look at the PE absolute vorticity, which we shall denote r , which in cartesian co-ordinates is

$$r = f + \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$

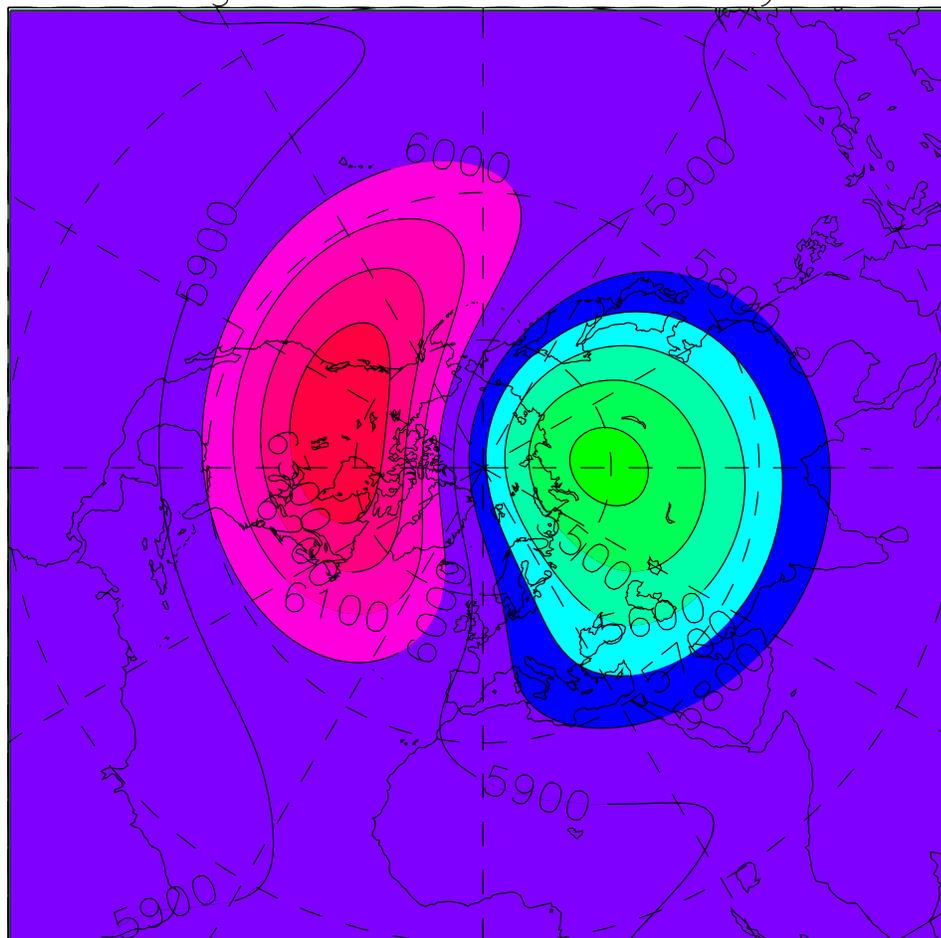
which is in terms of the full velocities and is available from both models. Figure 4.14 shows r for the PE model after 5 days and figure 4.15 shows the same field for the SSG model. It is immediately apparent that there is large ridging in the contours in the SSG model, under and ahead of the high centre, which is not present in the PE model. Looking at the initial distribution of r , figure 4.16, we see that this ridging was not present in the initial data. If we look at q from the SSG model, figure 4.4, ridging can also be seen in a similar area to that in the r field but it is not as pronounced. This ridging represents the presence of low absolute vorticity fluid near the high centre and is consistent with the maintenance of the high centre at its original value. We conclude that in the SSG model there is either an advection of low absolute vorticity from low latitudes to maintain the feature, or there is some generation of low absolute vorticity ahead of the high centre. Since the absolute vorticity, either balanced or full, is not conserved in the SSG model, see section 2.2.2, either or both of these mechanisms may be responsible. Also there could be some generation due to the constraint enforcement and also due to the correction step. As neither q nor r is a predictive variable in

the *SSG* model it is difficult to investigate this matter further. We could form the advection equation for q and use this as a diagnostic in trying to answer this question, but this has yet to be done. Since we do not have an analytic solution for the problem, one further question to be answered is whether or not the high in the *SSG* model is a true solution of the equations or not. The gain in height in the *SSG* model does increase as the timestep increases, as noted earlier, but as the timestep is decreased the solution converges to one where the height field still does not reduce from its original value. This, coupled with the same results being obtained from different advection schemes, suggests that the approximations used in the corrector step can lead to a slight growth in high values of height. However the underlying solution is that the height field does not weaken as it does in the *PE* model. This suggests that the *SSG* model is more insistent on retaining highs than the *PE* model.

4.2.2 The Rossby-Haurwitz Problem

Known solutions to the shallow-water primitive equations on the sphere are uncommon, with the most well-known being the Rossby-Haurwitz wave, see Haurwitz [33]. This wave will propagate from east to west at a known angular velocity without changing shape, if the equations are non-divergent. However, for the shallow-water equations with a free surface this is not the case. The propagation speed is reduced, the shape is no longer unchanged, and the wave is unstable for wave-numbers greater than 5, Hoskins [37]. This problem is still a standard test problem for the primitive shallow-water equations on the sphere and has been studied by many authors, for example, Phillips [57], Cullen [14], and Doron et al. [25]. The problem is determined by four parameters; a scale height, which is simply a base height for the free surface, the wave number which we are interested in, a parameter ω which sets an underlying zonal wind field from west to east, and a parameter K which controls the amplitude of the wave. Complete details

PE Model. MB89 Data
Height Field after 5 days.



5300 5500 5700 5900 6100 6300 6500

Figure 4.13: Timestep=1800 seconds.

PE Model. MB89 Data
P.E. r Field after 5 days.

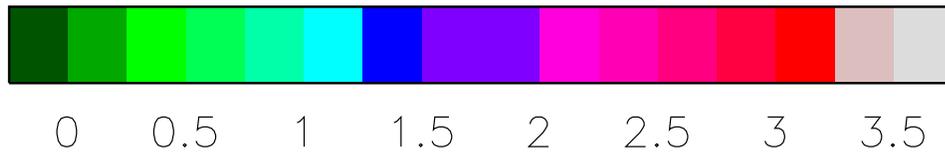
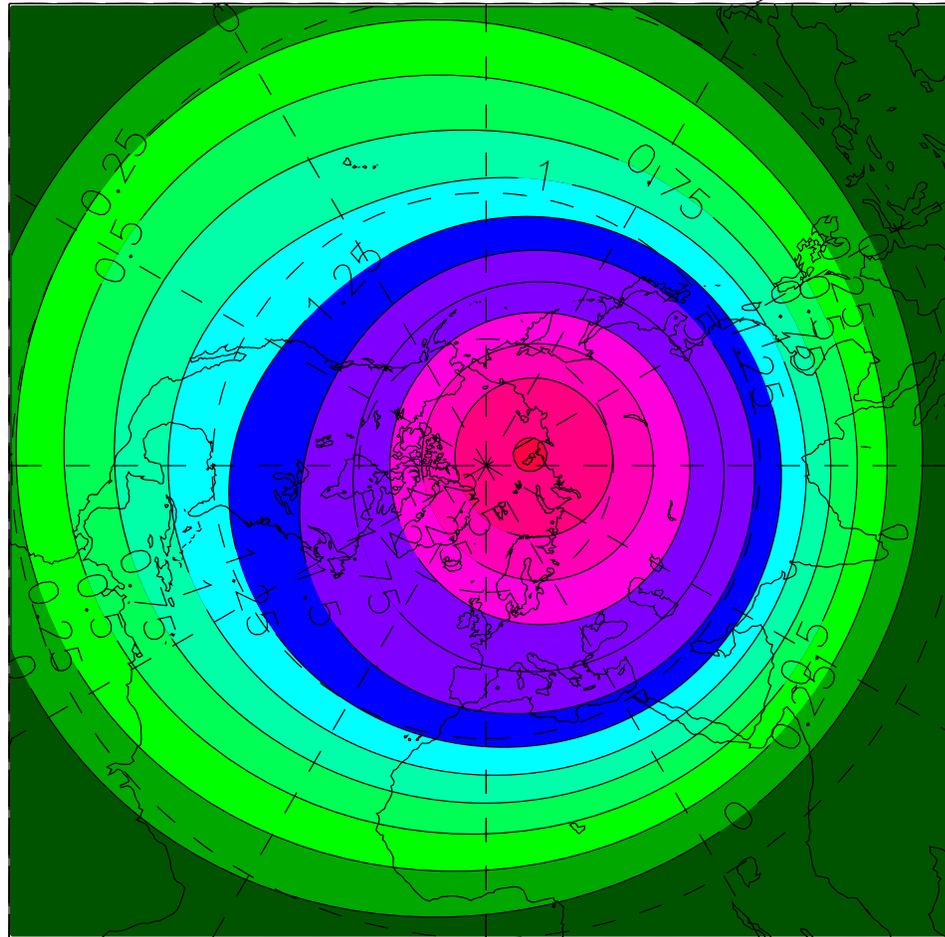


Figure 4.14: Timestep=1800 seconds. Contour interval = 0.5×10^{-8} .

QE Model. Semi-Lagrangian MB89 Data
P.E. r Field after 5 days.

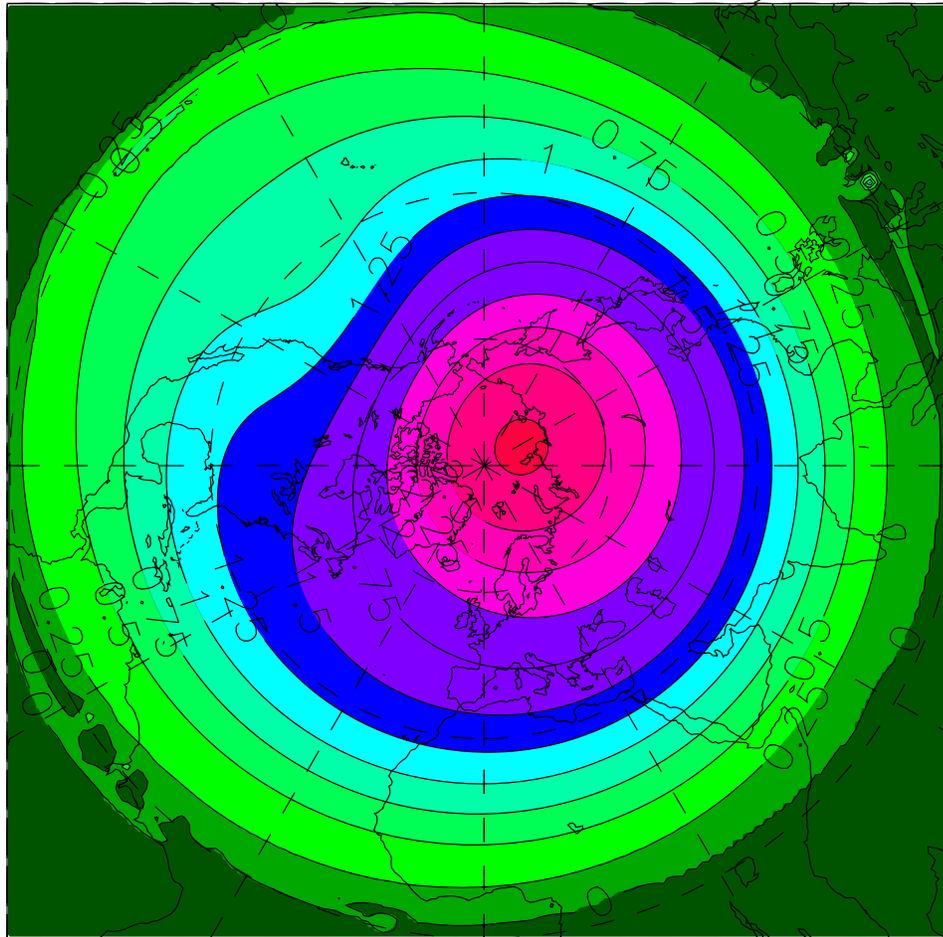


Figure 4.15: Timestep=1800 seconds. Contour interval = 0.5×10^{-8} .

QE Model. MB89 Data
Initial P.E. r Field.

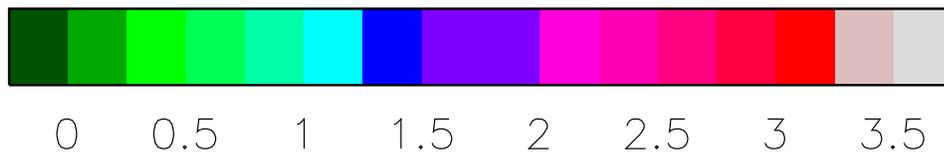
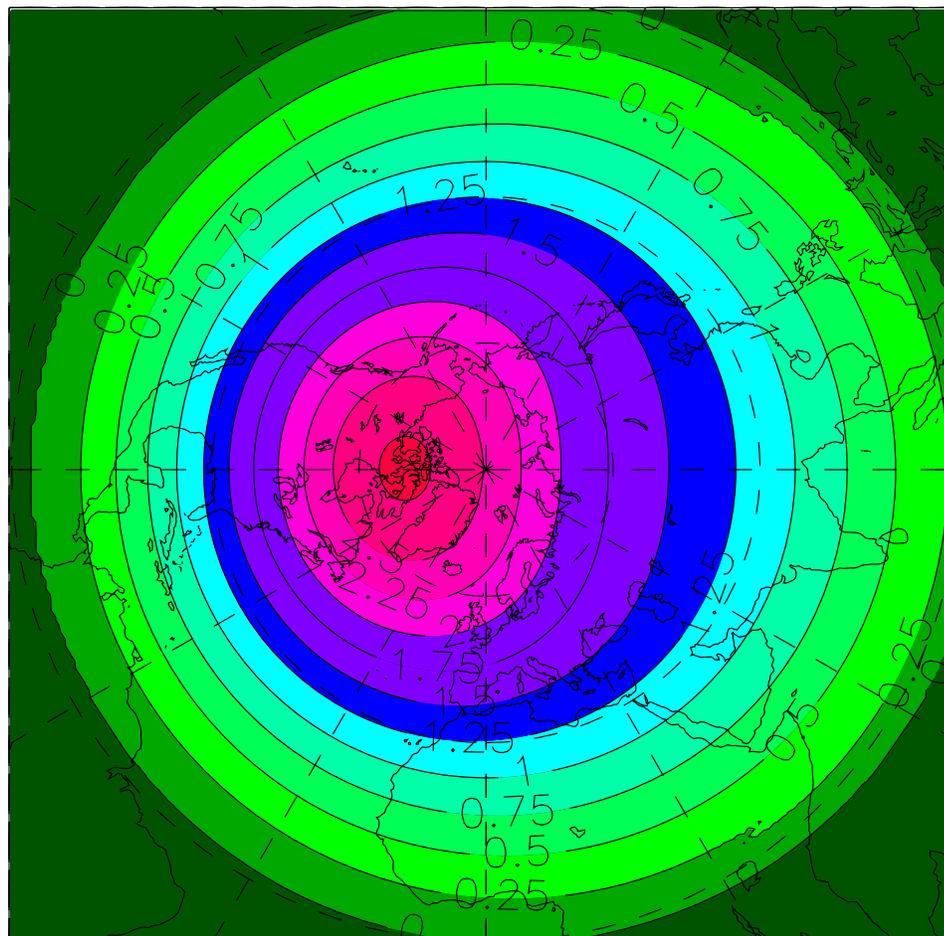


Figure 4.16: Contour interval = 0.5×10^{-8} .

can be found in Phillips [57]. The standard problem used for wave-number 4 is found in Phillips and we would like to use this a test problem for the SSG model. However, this standard problem has areas of negative absolute vorticity in the initial data. In fact no true Rossby-Haurwitz wave can be used as the initial conditions for the SSG model because the data requires that a non-zero height gradient exists along the equator, see figure 4.19 for the Rossby-Haurwitz wave initial height data for the parameters; wave number 4, $\omega = 7.848 \times 10^{-7}$, $K = 7.848 \times 10^{-7}$ and a scale height, referred to as ϕ_0 in Phillips, of $5500 \times g$. For small values of ω and K the initialisation procedure of the SSG model can be used without affecting the initial data extensively, the initialised height field for the parameters just given is shown in figure 4.17. We run the SSG model from this initialised field and the PE model from the analytic initial data using a timestep of 1800 seconds at 192×129 resolution. Running the PE model from the initialised field used as the starting field for the SSG model produces a solution almost identical to that obtained by running the PE model from the analytic initial data. Figure 4.18 shows the height field from the SSG model after 5 days. The field is largely unaltered with the axes of the troughs and ridges showing a slight westwards tilt which is typical of solutions gained with the primitive equations. A similar result is gained for the primitive equations run from the true initial data; figure 4.20 shows the height field after 5 days. The main difference between the two solutions is the distance the wave has propagated to the west. In the SSG model the wave has propagated approximately 60 degrees whilst in the PE model it has propagated about 96 degrees. The theoretical distance the wave should have propagated in a non-divergent primitive equation model is 101 degrees for the parameters chosen here. To understand this propagation behaviour we perform a linearised analysis of the non-divergent SSG equations for travelling waves following Shutts [68], where in section 3.1 in his paper he performs the analysis for the planetary semi-geostrophic equations. The linearised non-divergent SSG

equations are,

$$\frac{\partial u_0}{\partial t} - 2\Omega v \sin\phi = \frac{-1}{a \cos\phi} \frac{\partial \Phi}{\partial \lambda} \quad (4.1)$$

$$\frac{\partial v_0}{\partial t} + 2\Omega u \sin\phi = \frac{-1}{a} \frac{\partial \Phi}{\partial \phi} \quad (4.2)$$

$$\frac{1}{\cos\phi} \left[\frac{\partial u}{\partial \lambda} + \frac{\partial v \cos\phi}{\partial \phi} \right] = 0 \quad (4.3)$$

$$u_0 \sin\phi = -\frac{1}{2\Omega a} \frac{\partial \Phi}{\partial \phi} \quad (4.4)$$

$$v_0 \sin\phi = \frac{1}{2\Omega a \cos\phi} \frac{\partial \Phi}{\partial \lambda} \quad (4.5)$$

where $\Phi = gh$. Equations (4.1) and (4.2) can be used to give expressions for u and v in terms of Φ using equations (4.4) and (4.5) to substitute for u_0 and v_0 . Substituting these expressions for u and v into equation (4.3) gives a partial differential equation for Φ . Introducing the new variable

$$\mu = \sin\phi$$

and assuming that Φ is a travelling wave of form

$$\Phi = G_m(\mu) e^{i(m\lambda - \sigma t)}$$

then it can be shown that the partial differential equation for Φ can be written

$$\frac{(1 - \mu^2)}{\mu^2} \frac{\partial^2 G_m}{\partial \mu^2} - \frac{2}{\mu^3} \frac{\partial G_m}{\partial \mu} + \left(\frac{m\alpha_m}{\mu^2} - \frac{m^2}{\mu^2(1 - \mu^2)} \right) G_m = 0 \quad (4.6)$$

where

$$\alpha_m = -\frac{2\Omega}{\sigma}$$

This is the same equation as derived by Shutts, except for the factor of μ^2 in the denominator and a different α_m . The α_m we have derived here is identical to that for the primitive equations, and hence for linearised non-divergent flow waves that can be completely represented by both SSG and PE models should propagate with the same velocity. We note that the planetary semi-geostrophic

equations have propagation speeds which are in excess of those for the SSG and primitive equations, as shown by Shutts. The equal propagation speeds for the SSG and PE equation sets on an f -plane can be seen in the integrations of Whitaker [78]. However, as noted earlier, the Rossby-Haurwitz wave cannot be represented by the SSG equations. In chapter 2 we showed that any solution of the SSG equations at the equator must be of the form (2.23) and since Φ is assumed to be a solution it must satisfy

$$\Phi = d + c\phi^2 + \phi^{3+\delta}p(\lambda, \phi)$$

for some function $p \in C^2$ and $\delta > 0$. We now use the following expression for Φ

$$\Phi = \mu^{3+\delta}G_m(\mu)e^{i(m\lambda-\sigma t)}$$

since this then satisfies the condition at the equator. Substituting this form for Φ , and taking $\delta = 0$, gives

$$\mu(1 - \mu^2)\frac{\partial^2 G_m}{\partial \mu^2} + (4 - 6\mu^2)\frac{\partial G_m}{\partial \mu} + \left(\mu m \alpha_m - 6\mu - \frac{m^2 \mu}{(1 - \mu^2)} \right) G_m = 0 \quad (4.7)$$

which can be compared with equation (4.6) which was obtained with the original substitution. This is a generalised eigenvalue problem for α_m and solutions were obtained by solving the discrete version obtained by applying centred finite-differences to the derivative terms in equation (4.7). For wave-number $m = 4$, using 129 points to discretise the interval, all the eigenvalues are positive with the gravest mode corresponding to

$$\sigma = -\frac{4}{15}\Omega$$

which compares with the primitive equation gravest mode

$$\sigma = -\frac{2}{5}\Omega$$

We thus expect that the propagation speed of the wave in the SSG model should be two-thirds of that of the wave in the primitive equation model, when there is

no basic state zonal flow. For the parameters chosen for the numerical experiment the wave in the SSG model would have propagated 62 degrees in a non-divergent atmosphere, and this is in very good agreement with the 60 degrees that it propagated in the numerical experiment.

At this point we make some observations about the Rossby-Haurwitz wave as a simple representation of atmospheric motion. The atmosphere does not support the large amplitude oscillations at the equator required by the Rossby-Haurwitz wave and the propagation speed of the wave is much greater than that observed in the real atmosphere. The wave that can be represented by the SSG model is more typical of that seen in the real atmosphere, and the reduced propagation speed of the wave compared with the Rossby-Haurwitz wave is also much closer to the observed values. The exact agreement between primitive equation and SSG propagation speeds for waves un-constrained by the SSG solution at the equator is another desirable property of this equation set.

4.3 Real 500hpa Height Data

We now wish to consider the ability of the SSG model to handle real initial data to see if our model is capable of producing realistic large scale evolution on the whole sphere. We take as our initial field the 500 hpa height field from an operational forecast model. Once again we do not have an analytic solution for this problem; however we wish to see if the evolution of the features found in the initial data is physically realistic. Since there is no external forcing in this problem, it is essentially more useful as a test of the numerical scheme than as a validation of the equations. However, we expect that the motions with the longest time and space scales should persist since they are better resolved, whilst the shorter scale motions should be lost. As a guide to the “correct” solution to the problem we

QE Model. Rossby–Haurwitz Data
Initial Height Field.

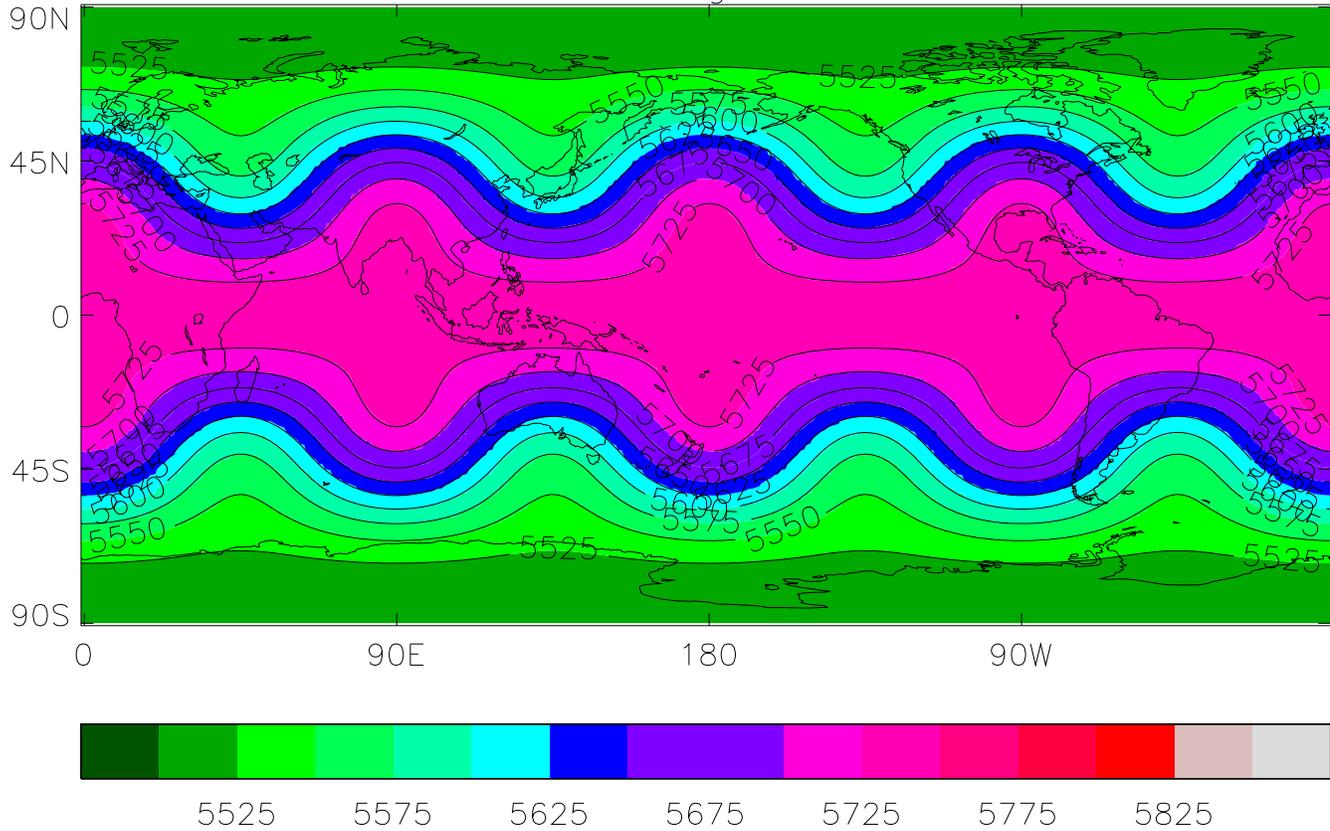


Figure 4.17:

QE Model. Rossby–Haurwitz Data
Height Field after 5 days.

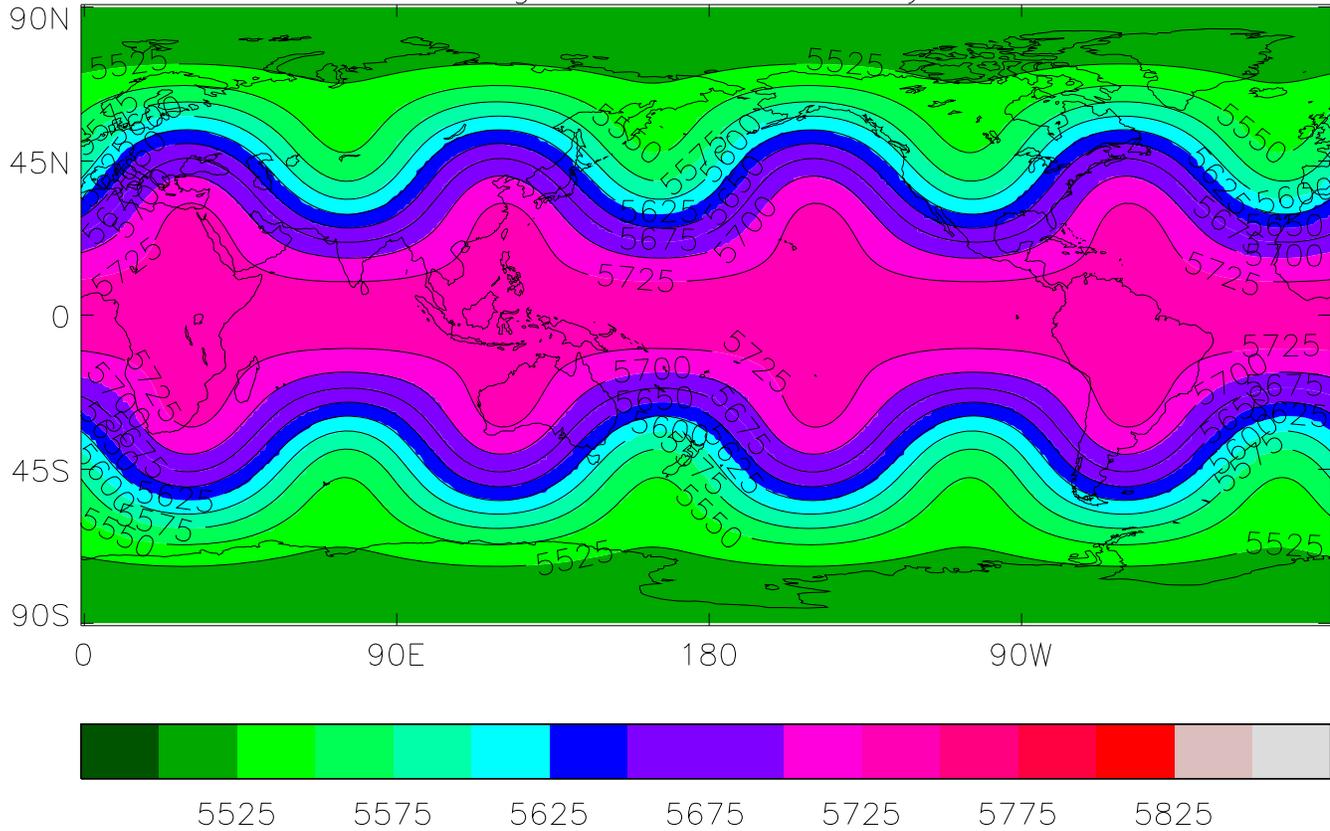


Figure 4.18: Timestep=1800 seconds.

PE Model. Rossby–Haurwitz Data
Initial Height Field.

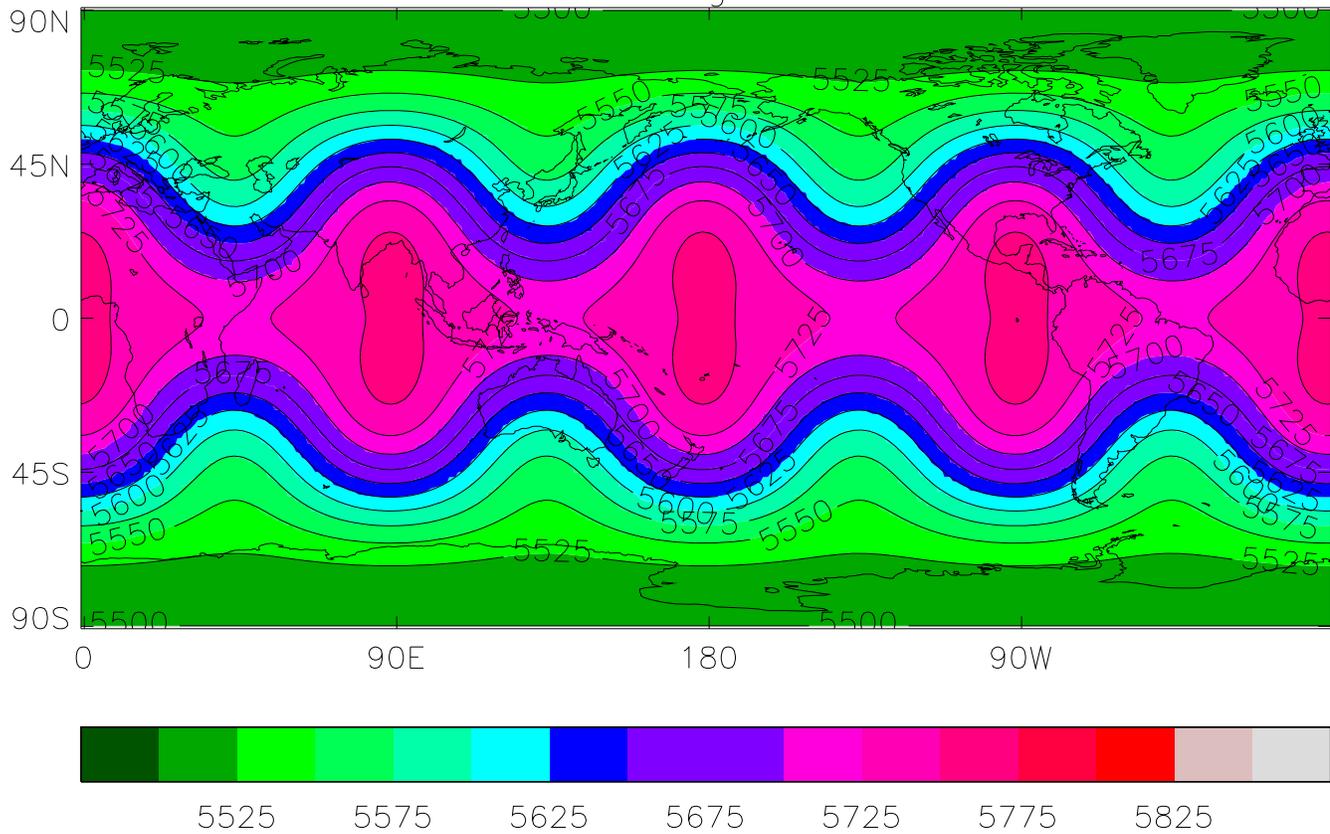


Figure 4.19:

PE Model. Rossby–Haurwitz Data
Height Field after 5 days.

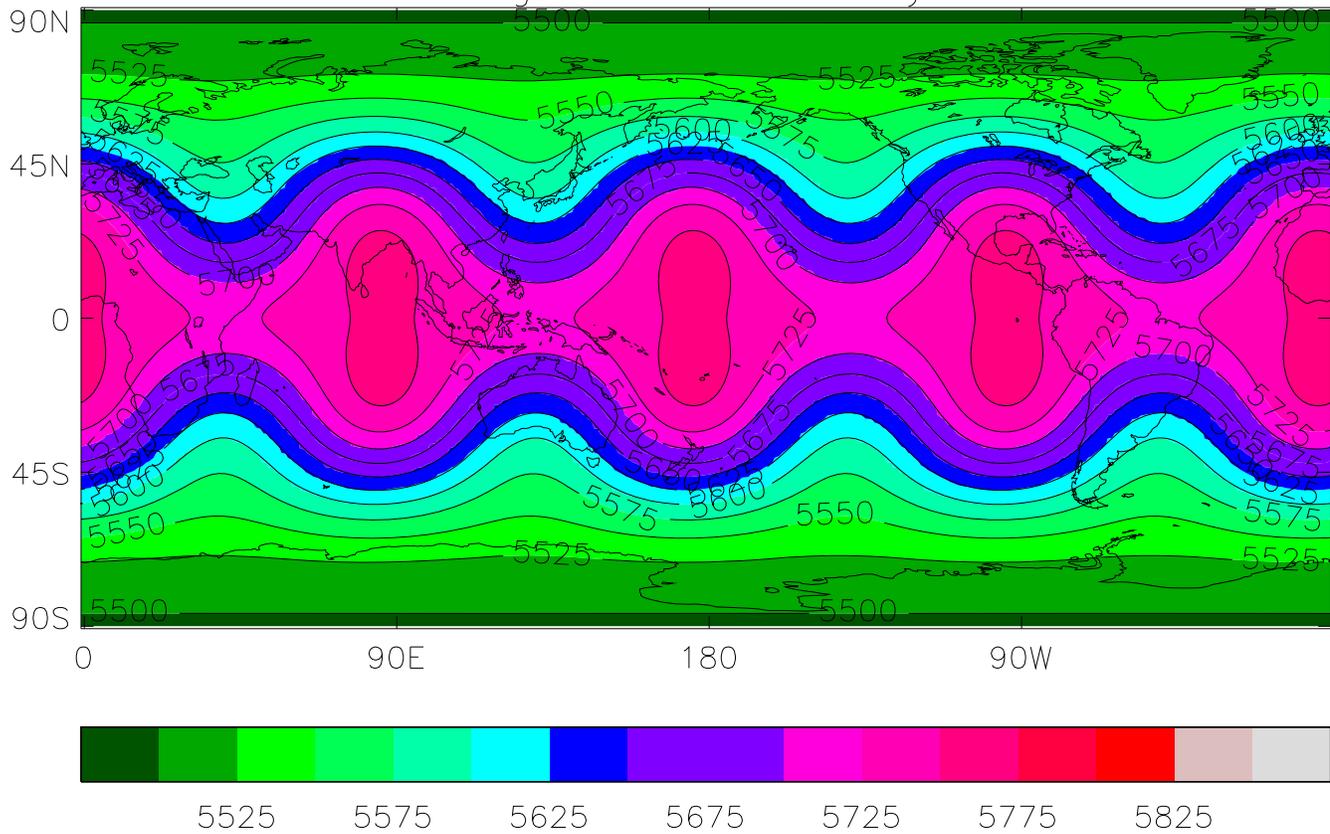


Figure 4.20: Timestep=1800 seconds.

perform the same integration for the primitive equations. We take as our initial data the 500hpa height field from the 1st February 1991, again at a resolution of 192×129 . The height field obtained after the SSG model initialisation procedure is shown in figure 4.21. This data contains a large high over northern Europe which has forced the usual zonal flow in this area to flow up towards the pole. This feature is known in meteorology as a blocking high and can persist for 2 to 3 weeks in the real atmosphere. We expect that the shallow-water model should maintain this feature whilst it will probably lose the smaller scale lows which are visible in the southern hemisphere. Figures 4.22 to 4.25 show the height field after 1, 3, 5, and 10 days respectively of an integration of the semi-Lagrangian SSG model starting from this initial data and using an 1800 second timestep. Looking at the first few days of the integration, it is noticeable how the smaller time-scale features are gradually disappearing whilst the main long time-scale feature, the blocking high over northern Europe, is largely unaltered. By day 5 this feature is the only significant one left and it persists throughout the ten days of the integration. This behaviour agrees with what we expected and to further verify this we run the PE model for the same case. We start the PE model from the SSG initial data after applying the PE initialisation to suppress any gravity waves. The application of this extra initialisation produces some changes in the initial data. Figure 4.26 shows the height field after 3 days of the integration and figure 4.27 after 10 days. The integrations of the two models after 3 days show very good agreement, better than that obtained for the idealised test cases. This suggests that the Lagrangian Rossby number, defined as

$$\frac{\left| \frac{D\mathbf{u}}{Dt} \right|}{|f\mathbf{u}|}$$

is small, much less than 1, for the flow, and hence that the geostrophic momentum approximation used to derive the SSG equations from the PE equations holds, see Hoskins [36]. As noted in section 4.2.2, both models have the same

phase speed for linearised non-divergent waves that are solutions of both equation sets. This suggests that the waves common in 500hpa flows are not of Rossby-Haurwitz type, and further enforces the view that the Rossby-Haurwitz wave is a poor representation of reality. The major areas of difference are in low latitudes where both models have formed high centres but in differing locations. This is not too surprising since the differences in the models are more pronounced in low latitudes where the PE model can support a wider variety of solutions than the SSG model. Comparing the features after 10 days, it is noticeable that the blocking high has been almost completely lost in the PE model whilst it is still prominent in the SSG model. Since there is no way of verifying the truth, we merely conclude that, as in the McDonald-Bates idealised data case, the SSG model maintains highs for longer than does the PE model. Other real data cases have been run but they add little or no extra information to that already given and so we have not included them here.

QE Model. 1/Feb/91 500hpa height Data
Initial Height Field.

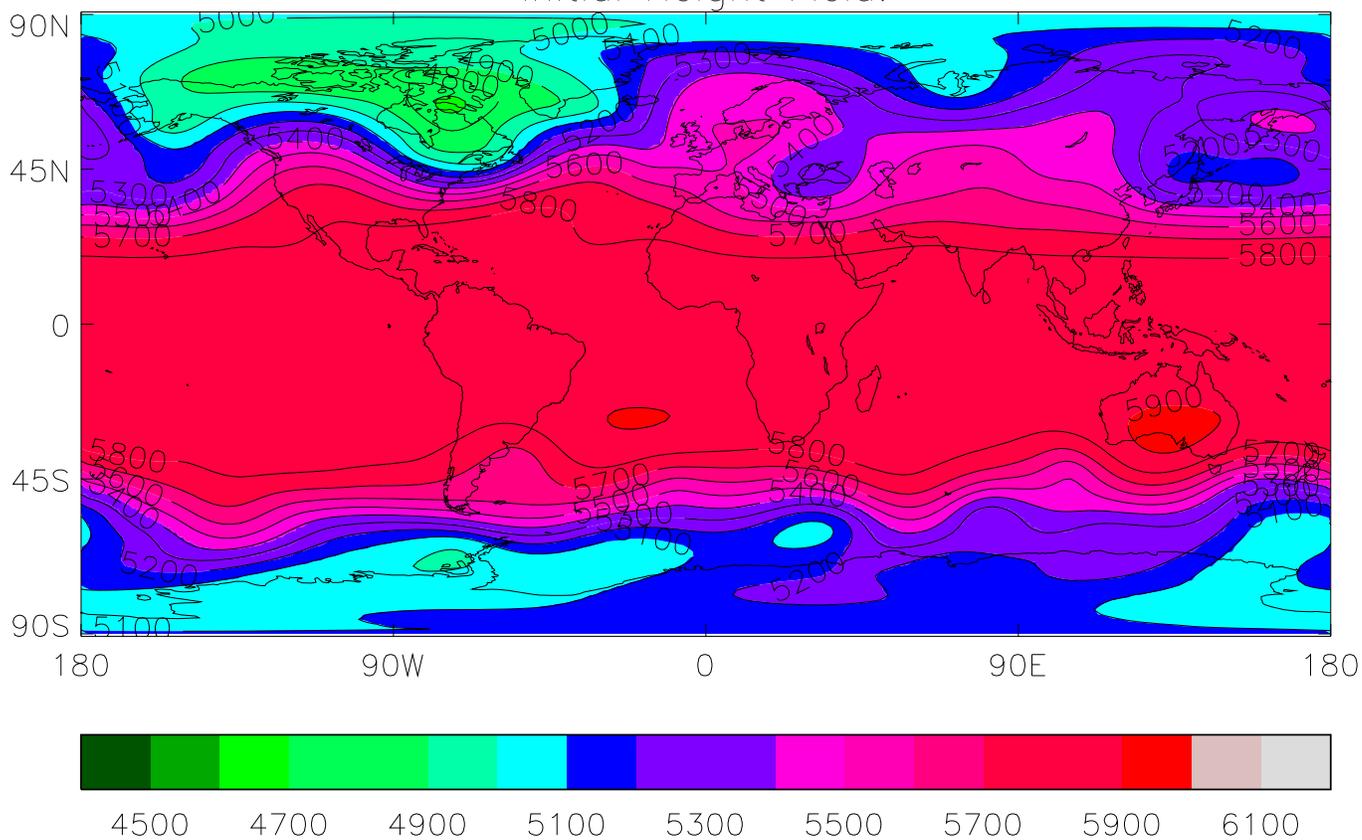


Figure 4.21:

QE Model. 1/Feb/91 500hpa height Data
Height Field after 1 day.

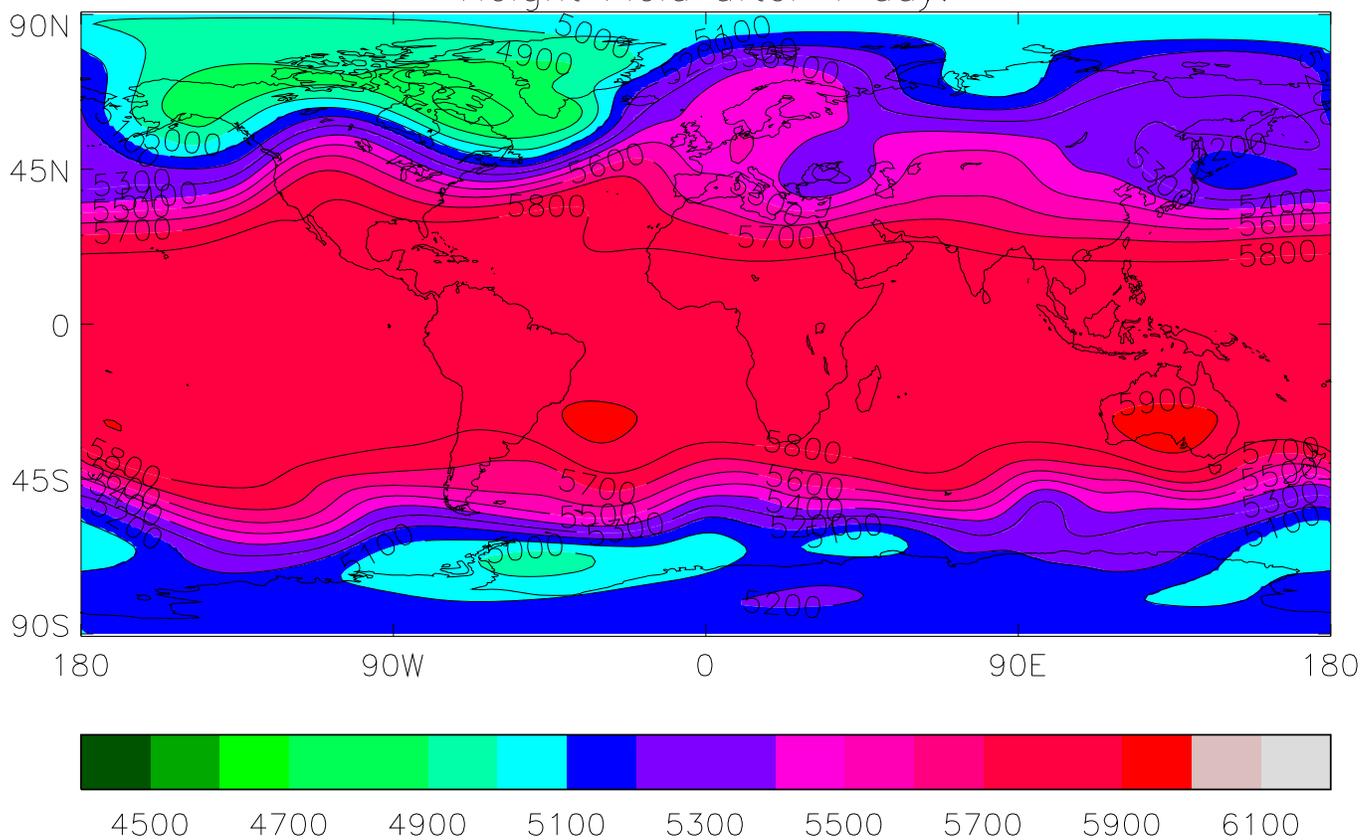


Figure 4.22: Timestep=1800 seconds.

QE Model. 1/Feb/91 500hpa height Data
Height Field after 3 days.

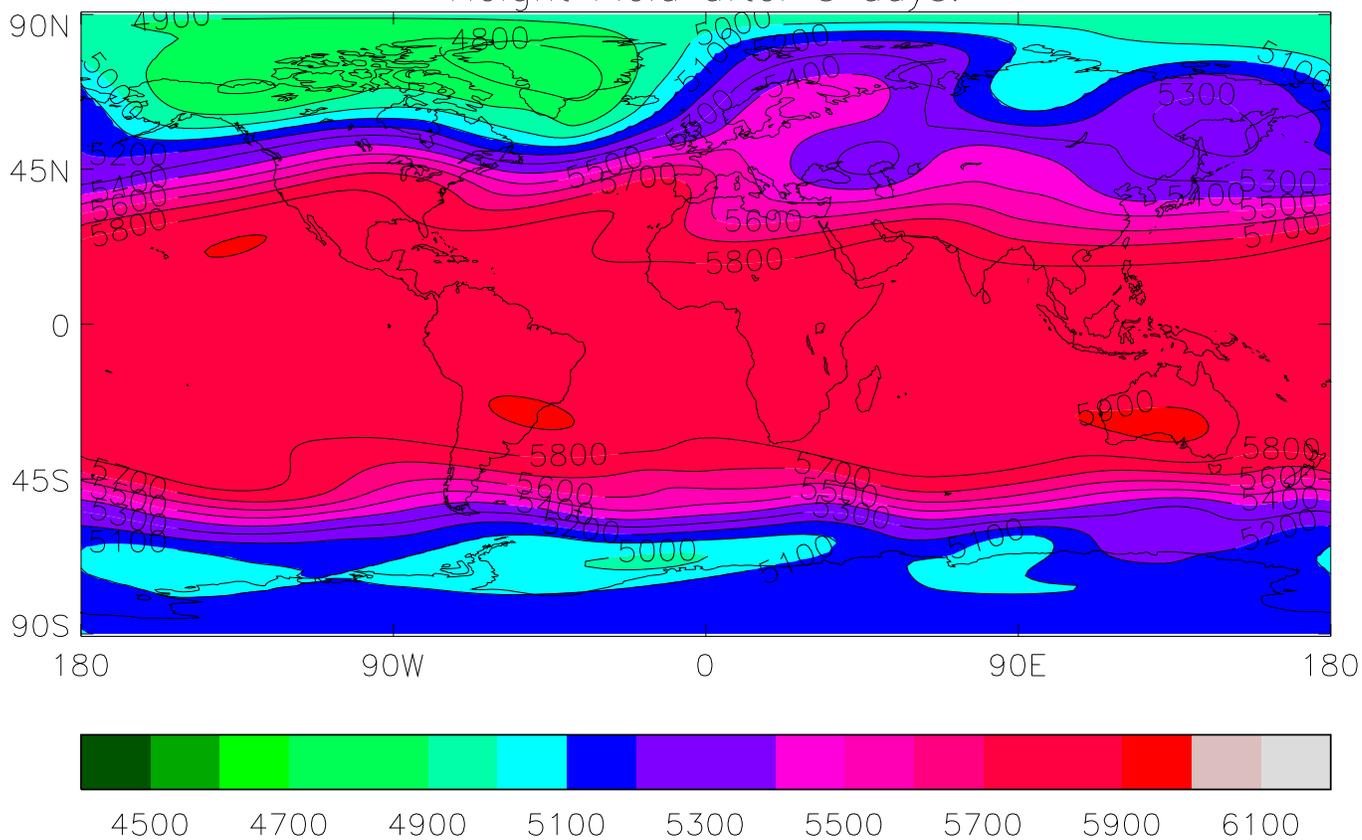


Figure 4.23: Timestep=1800 seconds.

QE Model. 1/Feb/91 500hpa height Data
Height Field after 5 days.

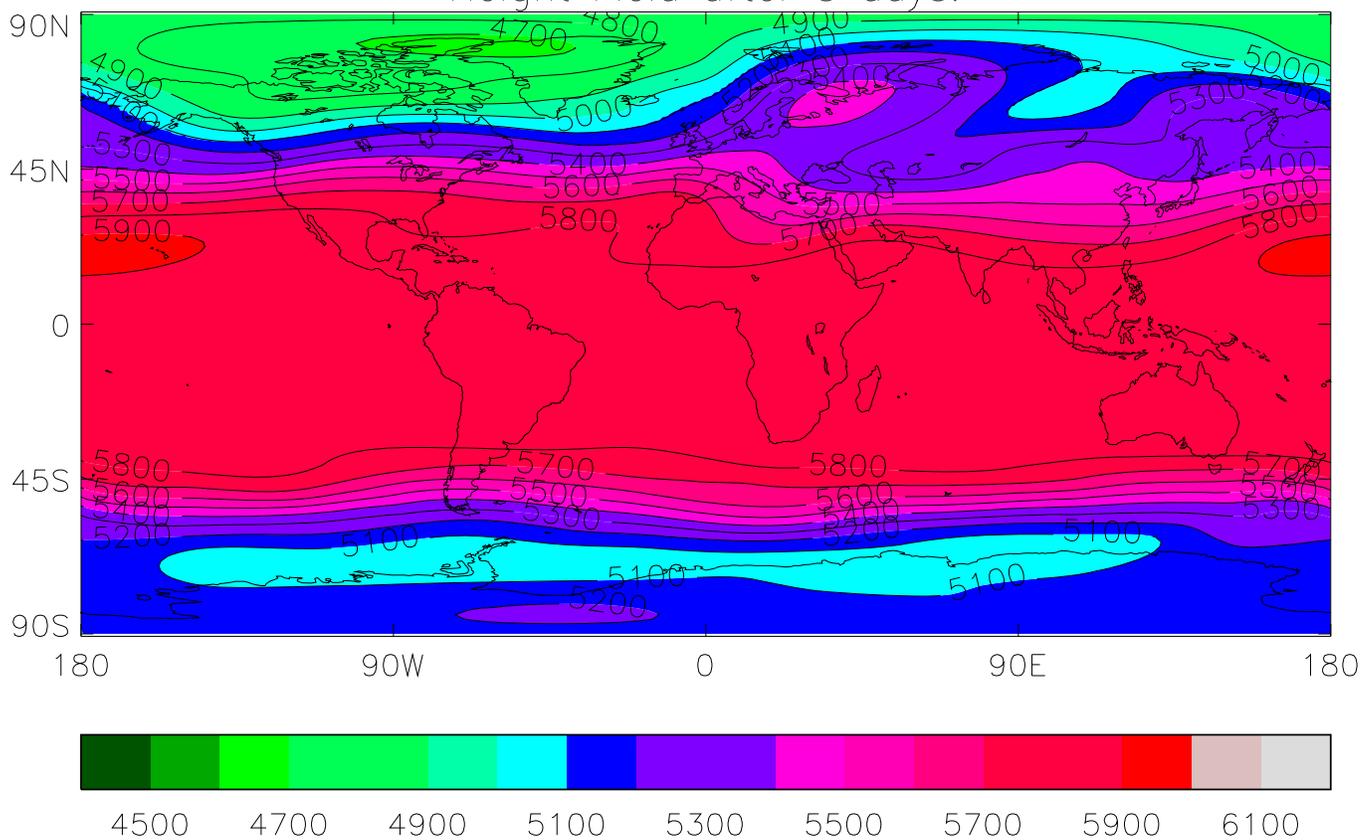


Figure 4.24: Timestep=1800 seconds.

QE Model. 1/Feb/91 500hpa height Data
Height Field after 10 days.

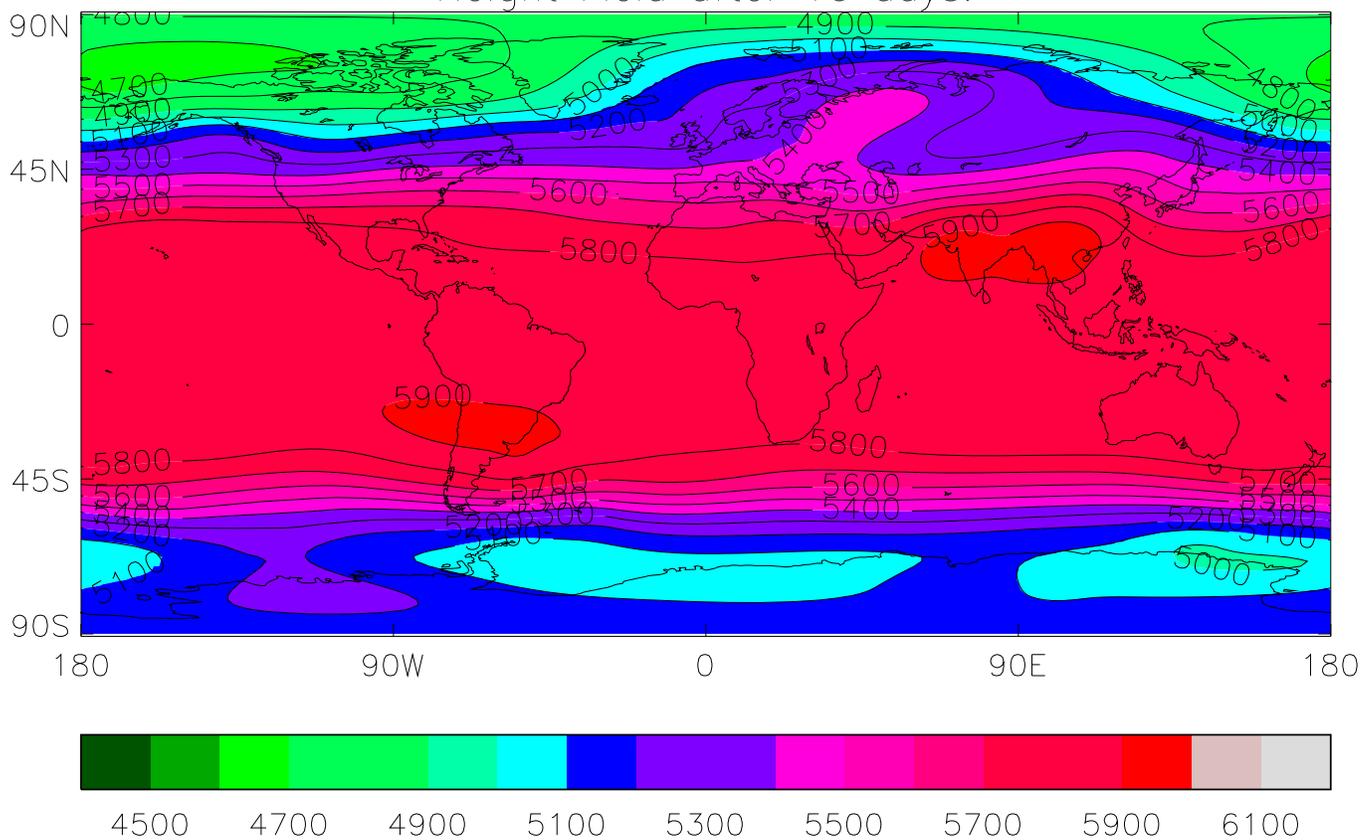


Figure 4.25: Timestep=1800 seconds.

PE Model. 1/Feb/91 500hpa height Data
Height Field after 3 days.

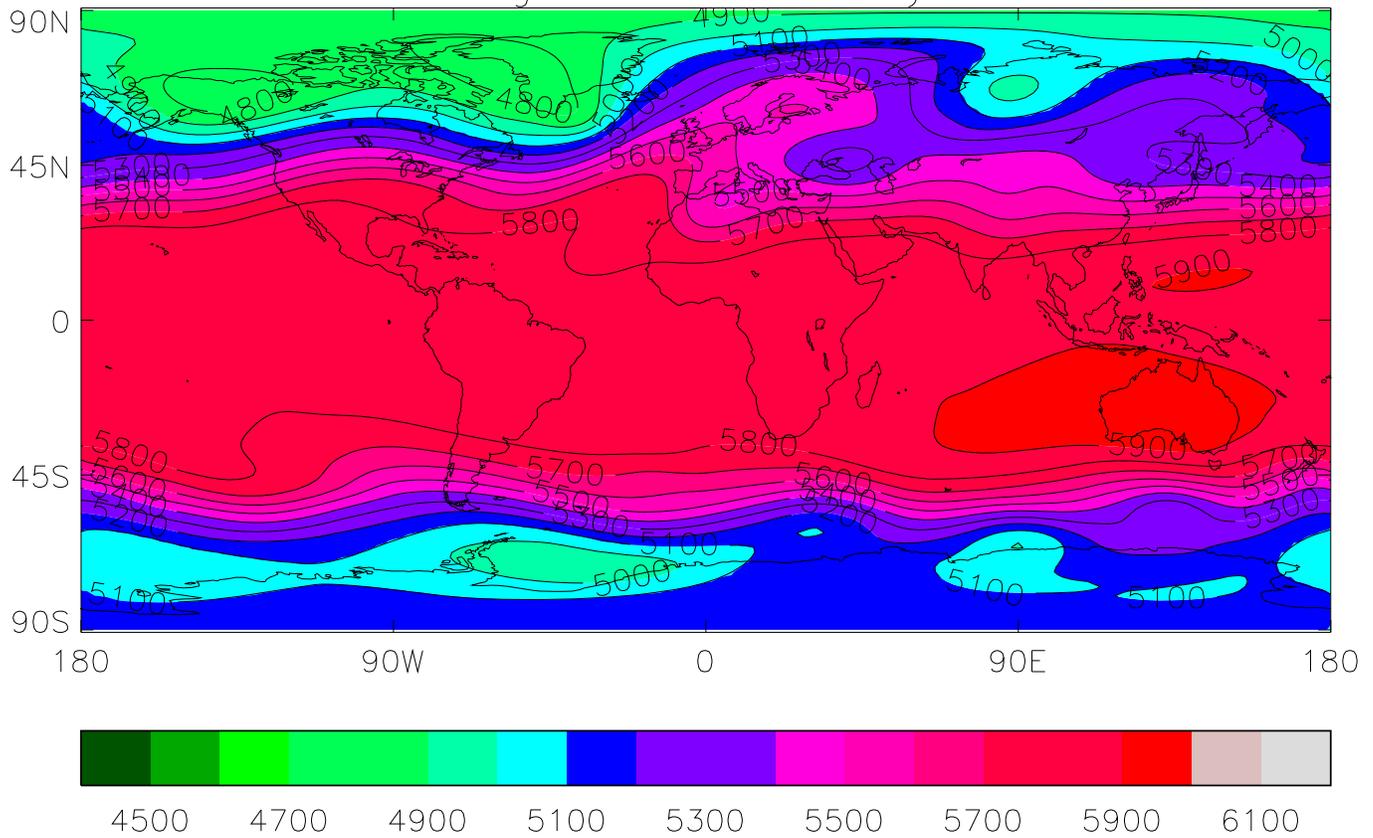


Figure 4.26: Timestep=1800 seconds.

PE Model. 1/Feb/91 500hpa height Data
Height Field after 10 days.

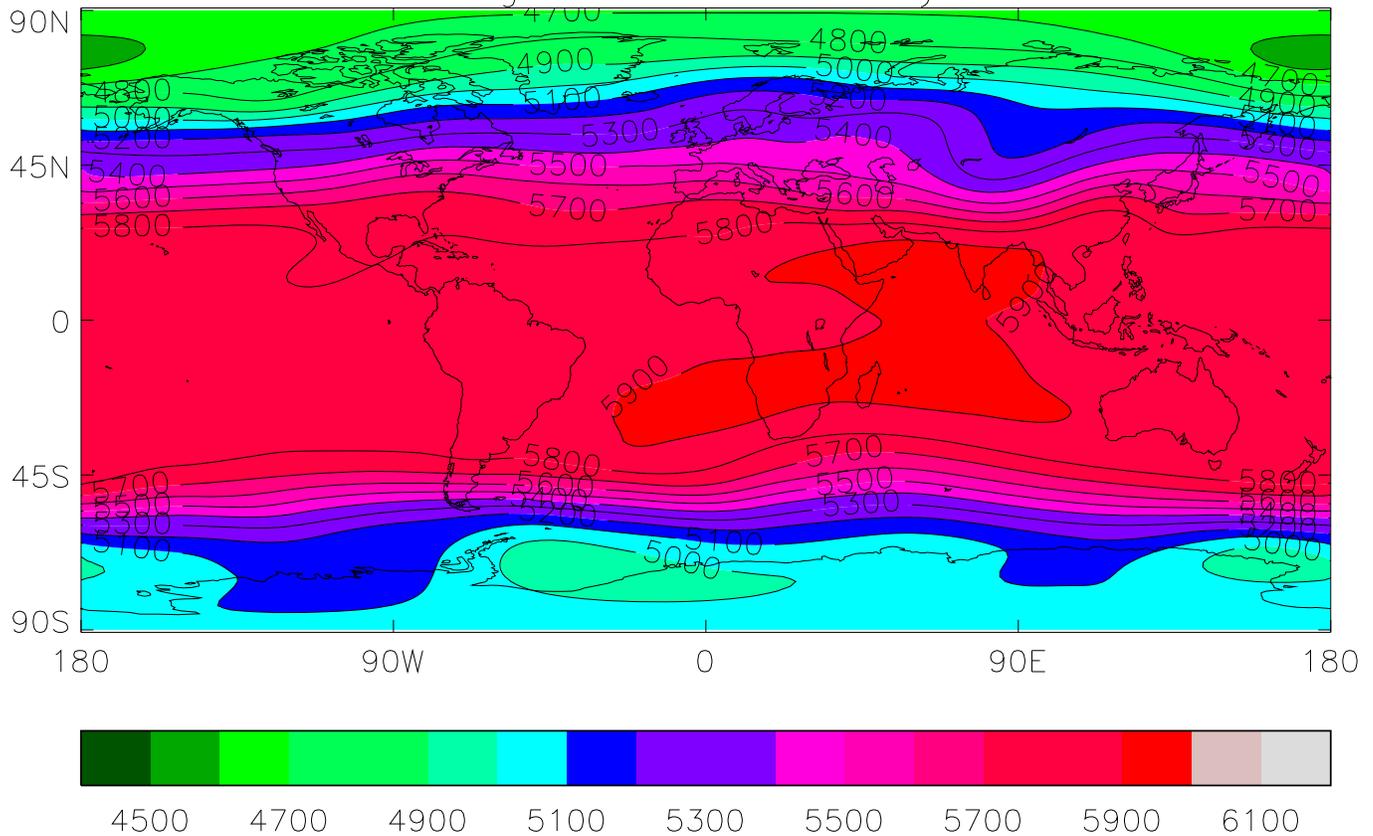


Figure 4.27: Timestep=1800 seconds.

Chapter 5

Conclusions and Discussion

We begin by drawing conclusions from the work in this thesis before embarking on a brief discussion of the extension of the work to three-dimensions on the sphere and making a few comments on the inclusion of friction in the equations.

5.1 Conclusions

- We have been unable to prove the existence of a unique solution to the SSG equations on the sphere given certain constraints. However we have shown how this might be proven given that certain key propositions, 2.2 and 2.5, can be proven.
- A predictor/corrector method has been developed which is robust and appears, in numerical experiments at fixed spatial resolution, to converge to the analytic solution. A necessary part of the procedure has been the construction of a method which restores any solution which leaves the solution set of the equations back into it. This part of the numerical procedure is one area where more work may be required to find the optimum scheme.

- The semi-Lagrangian advection scheme is better at preserving shape, does not suffer from small-scale noise, and has smaller errors than the Eulerian Heun advection schemes.
- Multigrid methods can be used to solve the elliptic equations which arise in the correction step in the numerical method. There is a problem with the transfer of boundary conditions between the various grids leading to slow convergence of the solution near the boundary, and thus a poor solution is obtained there compared with elsewhere in the domain. This does not appear to cause any problems in the numerical simulations so far attempted, but is an area where more work and understanding is required.
- Solutions from the Spherical Semi-Geostrophic equations compare well with those from the usual Primitive shallow-water equations for the test problems we have looked at. These test problems are primarily concerned with large scale motions and it remains to be seen how well the solutions compare for small scale motions on the sphere. The main difference in the solutions we have obtained lies in the SSG equations maintaining the intensity of high height values for longer. The results obtained from the SSG Rossby-Haurwitz simulations are more reminiscent of travelling waves in the real atmosphere, since they do not possess the large height gradients along the equator which are found in the usual Rossby-Haurwitz wave, and also their propagation speed is slower than that for the usual wave.

5.2 Discussion

5.2.1 Extension to 3 dimensions

Following Hoskins [36] the 3-dimensional SSG equations may be written in spherical polar co-ordinates (λ, ϕ, z) as

$$\frac{Du_0}{Dt} = S_p - \nabla_2 \phi - 2\Omega \underline{k} \times \underline{u} \quad (5.1)$$

$$\frac{D\theta}{Dt} = 0 \quad (5.2)$$

$$\frac{1}{a \cos \phi} \frac{\partial u}{\partial \lambda} + \frac{1}{a \cos \phi} \frac{\partial (v \cos \phi)}{\partial \phi} + \frac{\partial w}{\partial z} = 0 \quad (5.3)$$

$$\frac{g\theta}{\theta_{ref}} = \frac{\partial \phi}{\partial z} \quad (5.4)$$

$$\frac{g}{a \cos \phi} \frac{\partial \phi}{\partial \lambda} = f v_0 \quad (5.5)$$

$$\frac{g}{a} \frac{\partial h}{\partial \phi} = -f u_0 \quad (5.6)$$

where $\underline{u}_0 = (u_0, v_0)$, $\underline{u} = (u, v, w)$, θ_{ref} is a reference θ value,

$$S_p = \left(\frac{u v_0 \tan \phi}{a}, \frac{-u u_0 \tan \phi}{a} \right)$$

$$\nabla_2 = \left(\frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} + \frac{1}{a} \frac{\partial}{\partial \phi} \right)$$

and

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} + v \frac{1}{a} \frac{\partial}{\partial \phi} + w \frac{\partial}{\partial z}$$

A similar approach to that in chapter 2 can be used to reduce this set of equations to one equation for $\partial \phi / \partial t$. Equations (5.1) and (5.2) can be used to obtain expressions for u , v and w in terms of u_0 , v_0 , θ and ϕ . Equations (5.4)-(5.6) can now be used to substitute for u_0 , v_0 and θ in terms of ϕ and the resulting expressions for \underline{u} in terms of ϕ can be substituted into equation (5.3) to obtain a single equation for $\partial \phi / \partial t$. We can follow the same approach as used in chapter

2 to try to prove the existence of unique solutions to this equation under the generalised versions of the conditions required in the 2-d case. It is anticipated that this extension will not prove difficult if the results required to show existence and uniqueness for the shallow-water problem can be proven.

The numerical procedure can also be extended to 3-d. We have one more predictive equation for θ and a 3-d version of the correction equations. The main problem in extending the numerical scheme lies in the extension of the method to restore solutions outside the solution set back into it. It may be that application of the 2-d procedure on each model level in the vertical along with insisting that $\partial\theta/\partial z > 0$, the static stability condition, will suffice. However, this an area where some work may be required.

5.2.2 Inclusion of friction

Cullen [16] discussed the inclusion of friction into this equation set, and in shallow-water form the equations become

$$\frac{D\underline{u}_0}{Dt} = S_p - g\nabla h - 2\Omega\underline{k} \times \underline{u} - C_D\underline{u}_0 \quad (5.7)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\underline{u}) = 0 \quad (5.8)$$

$$g\frac{1}{a\cos\phi}\frac{\partial h}{\partial\lambda} = fv_0 - C_Du \quad (5.9)$$

$$g\frac{1}{a}\frac{\partial h}{\partial\phi} = -fu_0 - C_Dv \quad (5.10)$$

where

$$S_p = \left(\frac{uv_0\tan\phi}{a}, \frac{-uu_0\tan\phi}{a} \right)$$

and the Eulerian form of the derivative is

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u\frac{1}{a\cos\phi}\frac{\partial}{\partial\lambda} + v\frac{1}{a}\frac{\partial}{\partial\phi}$$

with C_D the drag coefficient. More complex functions representing frictional effects can readily be included. Assuming that the drag coefficient is constant then it can be shown that the rate of change of energy for this system is

$$-hC_D(\underline{u}_0^2 - \underline{u}_0 \cdot \underline{u} + \underline{u}^2)$$

which is negative definite. If we were to try to reduce this equation set to one equation in one variable we find that we cannot do it. This is because we are unable to replace \underline{u}_0 in the time derivatives of equation (5.7) with $\partial h / \partial t$ by using equations (5.9) and (5.10) without obtaining time derivatives of \underline{u} , which we cannot then eliminate.

We now consider an alternative way of including friction, namely,

$$\frac{D\underline{u}_0}{Dt} = S_p - g\nabla h - 2\Omega \underline{k} \times \underline{u} - C_D \underline{u} \quad (5.11)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\underline{u}) = 0 \quad (5.12)$$

$$g \frac{1}{a \cos \phi} \frac{\partial h}{\partial \lambda} = f v_0 - C_D u_0 \quad (5.13)$$

$$g \frac{1}{a} \frac{\partial h}{\partial \phi} = -f u_0 - C_D v_0 \quad (5.14)$$

where we have simply swapped \underline{u} and \underline{u}_0 where they appeared in the friction terms. We can reduce this equation set to one equation in one unknown since equations (5.13) and (5.14) can now be used to obtain expressions for \underline{u}_0 in terms of h without any terms in \underline{u} appearing. The rate of change of energy for this system is

$$hC_D((\underline{u}_0 - \underline{u})^2 - \underline{u}^2)$$

which can give rise to positive energy changes if \underline{u}_0 is larger than \underline{u} by a large enough amount. Any equation set which can produce positive energy sources is unsuitable for long time integrations. However, if there are no physical situations where positive energy could be generated then the equations could be used. It is

probably better to ask if there are any valid solutions to these equations which can give rise to a positive energy source. It is thus possible that this second system may be the better one to use, particularly as it could be considered as the more obvious of the two. This is because the first system used by Cullen is not the one obtained by simply using the Ekman velocity rather than the geostrophic wind in the geostrophic momentum approximation used to derive the equations. The system obtained that way is in fact this second system, for which the existence and uniqueness of solutions could be proved using the approach of Chapter 2 providing we can prove the outstanding results from Chapter 2. This is an area where more work may be useful.

Bibliography

- [1] H. H. Ahlberg, E. N. Nilson, J. L. Walsh. *The Theory of Splines and their Applications*. Academic Press 1967.
- [2] J. S. Allen, J. A. Barth and P. A. Newberger. *On intermediate models for barotropic continental shelf and slope field flows: Part 1, formulation and comparison of exact solutions*. J. Phys. Oceanogr. **20** pp 1949–1973, 1990.
- [3] A. Arakawa. *Lecture notes on UCLA general circulation model*. Department of Meteorology, UCLA, 1971.
- [4] A. Arakawa. *Geostrophic adjustment in continuous and discrete cases*. World Met. Organisation Program in Short- Medium-Range Weather Prediction Research **No 8** pp 24–41, 1984.
- [5] A. Arakawa and V. R. Lamb. *Computational design of the basic dynamical processes of the UCLA general circulation model*. In *Methods of Computational Physics* **Vol 17** pp 174–265, Academic Press 1977.
- [6] S. R. M. Barros. *Multigrid methods for Two- and Three-Dimensional Poisson-Type equations on the sphere*. J. Comp. Phys. **92** pp 313–348, 1991.
- [7] J. R. Bates, F. H. M. Semazzi, R. W. Higgins and S. R. M. Barros. *Integration of the shallow water equations on the sphere using a semi-Lagrangian scheme with a multigrid solver*. Mon. Wea. Rev. **118** pp 1615–1627, 1990.

- [8] W. L. Briggs, *A Multigrid tutorial*. SIAM 1987.
- [9] E. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw-Hill 1955.
- [10] F. Castelli, R. L. Bras and K. A. Emanuel. *An analytic approach to the non-linear dynamics of moist frontogenesis*. J. Atmos. Sci. **50** pp 1504–1518, 1993.
- [11] J. G. Charney. *The dynamics of long waves in a baroclinic current*. J. Meteor. **4** pp 135–163, 1947.
- [12] J. G. Charney, R. Fjortoft and J. von Neumann. *Numerical integration of the barotropic vorticity equation*. Tellus, **2** pp 237–254, 1950.
- [13] S. Chynoweth, D. Porter and M. J. Sewell. *The parabolic umbilic and atmospheric fronts*. Proc. R. Soc. Lond. **419** pp 337–362, 1988.
- [14] M. J. P. Cullen. *Integrations of the primitive equations on a sphere using the finite element method*. Quart. J. Roy. Met. Soc. **100** pp 555–562, 1974.
- [15] M. J. P. Cullen. *Implicit Finite-Difference Method for Modelling Discontinuous Atmospheric Flows*. J. Comp. Phys. **81** pp 319–348, 1989.
- [16] M. J. P. Cullen. *On the incorporation of atmospheric boundary layer effects into a balanced model*. Quart. J. Roy. Met. Soc. **115** pp 1109–1131, 1989.
- [17] M. J. P. Cullen, S. Chynoweth and R. J. Purser. *On semi-geostrophic flow over synoptic scale topography*. Quart. J. Roy. Met. Soc. **113** pp 163–180, 1987.
- [18] M. J. P. Cullen and T. Davies. *A Conservative Split-explicit Integration Scheme with 4th Order Horizontal Advection*. Quart. J. Roy. Met. Soc. **117** pp 993–1002, 1991.

- [19] M. J. P. Cullen, T. Davies and M. H. Mawson. *Conservative finite difference schemes for a unified forecast/climate model*. Unified Model Documentation Paper 10. Unpublished paper available from U.K. Meteorological Office.
- [20] M. J. P. Cullen and R. J. Purser. *An extended Lagrangian theory of semi-geostrophic frontogenesis*. J. Atmos. Sci. **41** pp 1477–1497, 1984.
- [21] M. J. P. Cullen and R. J. Purser. *Properties of the Lagrangian Semi-geostrophic Equations*. J. Atmos. Sci. **46** pp 2684–2697, 1989.
- [22] M. J. P. Cullen, J. Norbury, and R. J. Purser. *Generalized Lagrangian Solutions for Atmospheric and Oceanic Flows*. SIAM J. Appl. Maths. **51** pp 20-31, 1991.
- [23] M. J. P. Cullen, J. Norbury, R. J. Purser, and G. J. Shutts. *Modelling the quasi-equilibrium dynamics of the atmosphere*. Quart. J. Roy. Met. Soc. **113** pp 735–758, 1987.
- [24] R. Daley. *Atmospheric data analysis*. Cambridge university press, 1991.
- [25] E. Doron, A. Hollingsworth, B. J. Hoskins, and A. J. Simmons. *A comparison of grid-point and spectral methods in a meteorological problem*. Quart. J. Roy. Met. Soc. **100** pp 371–383, 1974.
- [26] C. C. Douglas and J. Douglas, Jr. *A unified convergence theory for abstract multigrid and multilevel algorithms, serial and parallel*. SIAM J. Numer. Anal. **30** pp 136–158, 1993.
- [27] A. Eliassen. *The quasi-static equations of motion*. Geofys. Publ. **17** No. 3, 1948.
- [28] S. R. Fulton, P. E. Ciesielski and W. H. Schubert. *Multigrid methods for Elliptic Problems: A Review*. Mon. Wea. Rev. **115** pp 942–959, 1986.

- [29] J. Gary, S. McCormick and R. Sweet. *Successive Over-Relaxation, Multigrid, and Preconditioned Conjugent Gradient Algorithms for solving a Diffusion Problem on a Vector Computer*. Appl. Maths. and Comp. **13** pp 285–309, 1983.
- [30] P. R. Gent and J. C. McWilliams. *Consistent balanced models in bounded and periodic domains*. Dyn. Atmos. Oceans **7** pp 67–93, 1983.
- [31] P. R. Gent and J. C. McWilliams. *Regimes of validity for balanced models*. Dyn. Atmos. Oceans **7** pp 167–183, 1983.
- [32] W. Hackbusch. *Multigrid methods and applications*. Springer-Verlag, Berlin, 1985.
- [33] B. Haurwitz. *The motion of atmospheric disturbances on the spherical earth*. J. Marine Res. **3** pp 254–267, 1940.
- [34] M. W. Holt. *Semi-geostrophic moist frontogenesis in a Lagrangian model*. Dyn. Atmos. Oceans **14** pp463–481, 1990.
- [35] M. W. Holt and A. J. Thorpe. *Localized forcing of slantwise motion at fronts*. Quart. J. Roy. Met. Soc. **117** pp 943–963, 1991.
- [36] B. J. Hoskins. *The geostrophic momentum approximation and the semi-geostrophic equations*. J. Atmos. Sci. **32** pp 233–242, 1975.
- [37] B. J. Hoskins. *Stability of the Rossby-Haurwitz wave*. Quart. J. Roy. Meteor. Soc. **99**, pp 723–745, 1973.
- [38] B. J. Hoskins and F. P. Bretherton. *Atmospheric frontogenesis models: mathematical formulation and solution*. J. Atmos. Sci. **29** pp 11–37, 1972.

- [39] B. J. Hoskins, M. E. McIntyre and A. W. Robertson. *On the use and significance of isentropic potential vorticity maps*. Quart. J. Roy. Meteor. Soc. **111** pp 877–946, 1985.
- [40] J. Linden, B. Steckel and K. Stüben. *Parallel multigrid solution of the Navier-Stokes equations on general 2-D domains*. Parallel Computing **7**, 1988.
- [41] P. Lynch and Xiang-Yu Huang. *Initialisation of the HIRLAM model using a digital filter*. Mon. Wea. Rev. **120** pp 1019–1034, 1992.
- [42] A. McDonald. *A semi-Lagrangian and semi-implicit two-time-level integration scheme*. Mon. Wea. Rev. **114** pp 824–830, 1986.
- [43] A. McDonald and J. R. Bates. *Improving the estimate of the departure point position in a two-time level semi-Lagrangian and semi-implicit scheme*. Mon. Wea. Rev. **115** pp 737–739, 1987.
- [44] A. McDonald and J. R. Bates. *Semi-Lagrangian integration of a gridpoint shallow-water model on the sphere*. Mon. Wea. Rev. **117** pp 130–137, 1989.
- [45] J. C. McWilliams and P. R. Gent. *Intermediate models of planetary circulations in the atmosphere and ocean*. J. Atmos. Sci. **37** pp 1657–1678, 1980.
- [46] L. G. Margolin and J. J. Pyun. *A method for treating hourglass patterns*. Proc. 5th International conference on Numerical methods in Laminar and Turbulent flow, C. Taylor, W. Habashi and M. Hafez editors. Montreal, Canada 1987.
- [47] B. L. Marshall. *Tests of the Heun advection scheme*. Met O. 11 Technical Note No 31. 1989. Unpublished document available from U.K. Meteorological Office.

- [48] M. H. Mawson and M. J. P. Cullen. *An idealised simulation of the Indian monsoon using primitive equation and quasi-equilibrium models*. Quart. J. Roy. Met. Soc. **118** pp 153–164, 1992.
- [49] F. Mesinger. *Horizontal advection schemes on a staggered grid - An enstrophy and energy conserving model*. Mon. Wea. Rev. **109** pp 467–478, 1981.
- [50] F. Mesinger and A. Arakawa. *Numerical methods used in atmospheric models. Volume 1*. W.M.O GARP Publication series No 17, Geneva 1976.
- [51] R. H. Miller. *A horror story about integration methods*. J. Comp. Phys. **93** pp 469–476, 1991.
- [52] C. Miranda. *Partial Differential Equations of Elliptic Type*. Second revised edition, Springer-Verlag 1970.
- [53] S. Mizohata. *Theory of Partial Differential Equations*. Cambridge University Press 1973.
- [54] S. Moorthi and R.W. Higgins. *Application of Fast Fourier Transforms to the Direct Solution of a Class of Two-Dimensional Separable Elliptic Equations on the Sphere*. Mon. Wea. Rev. **121** pp 290–296, 1993.
- [55] N.H. Naik and J. Van Rosendale. *The improved robustness of Multigrid elliptic solvers based on multiple semicoarsened grids*. SIAM J. Numer. Anal. **30** pp 215–229, 1993.
- [56] J. Pedlovsky. *Geophysical Fluid Dynamics*. 2nd Ed. Springer-Verlag 1987.
- [57] N. A. Phillips. *Numerical integration of the primitive equations on the sphere*. Mon. Wea. Rev. **87** pp 333-345, 1959.
- [58] R. T. Pierrehumbert. *Stratified semi-geostrophic flow over two-dimensional topography in an unbounded domain*. J. Atmos. Sci. **42** pp 523–526, 1985.

- [59] L. F. Richardson. *Weather prediction by numerical process*. Cambridge University Press, 236pp. 1922. Reprinted by Dover.
- [60] C. A. Riegel and A. F. C. Bridger. *Fundamentals of atmospheric dynamics and hydrodynamics*. World Science 1992.
- [61] A. Robert. *A stable numerical integration scheme for the primitive meteorological equations*. Atmos. Ocean. **19** pp 35–46, 1981.
- [62] G. Robinson. *Fluid dynamics in parallel using an unstructured mesh*. First International Conference on Parallel Processing for Computational Mechanics, Southampton, 1990.
- [63] R. Salmon. *New equations for nearly geostrophic flow*. J. Fluid Mech. **153** pp 461–477, 1985.
- [64] R. Salmon. *Semigeostrophic theory as a dirac-bracket projection*. J. Fluid. Mech. **196** pp 345–358, 1988.
- [65] R. Salmon. *What is potential vorticity ?* Woods Hole Oceanog. Inst. Tech. Rept., WHOI-93-24, pp 181–185, 1993.
- [66] T. I. Seidman. *Some “Complexity” Issues for Ill-Posed Problems*. Inverse Problems in Partial Differential Equations. Edited by D. Colton, R. Ewing and W. Rundell, SIAM 1990.
- [67] W. N. Shaw. *Weather forecasting*. Constable and Co. Ltd, London, 1913.
- [68] G. J. Shutts. *Planetary semi-geostrophic equations derived from Hamilton’s principle*. J. Fluid Mech. **208** pp 545–573, 1989.
- [69] G. J. Shutts and M. J. P. Cullen. *Parcel stability and its relation to semi-geostrophic theory*. J. Atmos. Sci. **44** pp 1318–1330, 1987.

- [70] J. Smagorinsky. *Global atmospheric modelling and the numerical simulation of climate*. in Weather and Climate modification, pp 633–686, edited by W. N. Hess, Wiley, New York, 1974.
- [71] D.M. Smith and S.P. Fiddes. *Efficient parallelisation of implicit and explicit solvers on a MIMD machine*. Information quarterly for high performance computing in engineering, No 2 pp 36–42, 1992 (an informal newsletter available from SERC Daresbury Lab., Warrington, England.)
- [72] A. Staniforth and J. Côté. *Semi-Lagrangian integration schemes for atmospheric models - A review*. Mon. Wea. Rev. **119** pp 2206–2223, 1991.
- [73] J. P. Van and G. D. Raithby. *Enhancements to the SIMPLE method for predicting incompressible fluid flows*. Numer. Heat Transfer **7** pp 147–163, 1984.
- [74] G. Veronis and E. A. Spiegel. *On the Boussinesq approximation for a compressible fluid*. Astrophysical Journal. **131** pp 442–447, 1960.
- [75] P. Welander. *Studies on the general development of motion in a two-dimensional idealised fluid*. Tellus **7** pp 141–156, 1955.
- [76] X. Wen and D. B. Ingham. *A new method for accelerating the rate of convergence of SIMPLE-like algorithms*. Int. J. Num. Meth. Fluids. **17** pp 385–400, 1993.
- [77] P. Wesseling. *An introduction to multigrid methods*. In Pure and Applied Mathematics Series, John Wiley and Sons, 1992.
- [78] J. S. Whitaker. *A comparison of primitive and balance equation simulations of baroclinic waves*. J. Atmos. Sci. **50** pp 1519–1530, 1993.

- [79] R. Wu and W. Blumen. *An analysis of Ekman boundary layer dynamics incorporating the geostrophic momentum approximation*. J. Atmos. Sci. **39** pp 1774–1782, 1982.
- [80] J. A. Young. *A theory for isallobaric flow in the planetary boundary layer*. J. Atmos. Sci. **30** pp 1584–1592, 1973.