DTIC FILE COPY (2)

CLaSSiC Project
Report CLaSSiC-87-05

October 1987

AD-A192 172

Adaptive Grid Refinement for Numerical Weather Prediction

William C. Skamarock

N00014-84-K-0267

DTIC
ELECTE
MAR 2 3 1988
S D
H

Center for Large Scale Scientific Computation
Building 460, Room 313
Stanford University
Stanford, California 94305

88 3 22 035

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

ADAPTIVE GRID REFINEMENT FOR NUMERICAL WEATHER PREDICTION

A DISSERTATION

SUBMITTED TO THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

COMPUTATIONAL GEOPHYSICAL FLUID DYNAMICS

By

William C. Skamarock

October ~~September~~ 1987

© Copyright 1988

by

William C. Skamarock

I certify that I have read this thesis and that in my opinion it is fully adequete, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Joseph Oliger
(Principal Advisor)
Department of Computer Science

I certify that I have read this thesis and that in my opinion it is fully adequete, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Robert L. Street
Department of Civil Engineering

I certify that I have read this thesis and that in my opinion it is fully adequete, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Joel H. Ferziger
Department of Mechanical Engineering

Approved for the University Committee on Graduate Studies:

Dean of Graduate Studies

Abstract

This dissertation describes the application of an adaptive solution technique to the dynamical equations used in numerical weather models. The adaptive technique employed is that of Berger and Oliger. It uses a finite difference method to integrate the dynamical equations first on a coarse grid and then on finer grids. The location of the fine grids is determined using a Richardson–type estimate of the truncation error in the coarse grid solution. By correctly coupling the integrations on the various grids, periodically re-estimating the error and recreating the finer grids, approximately uniformily accurate solutions are economically produced.

Two horizontally refining adaptive models, based on different sets of equations, are developed. The first, based upon the hydrostatic "primitive" equations of meteorology, is used to solve for the advection of a barotropic cyclone and to simulate the development of a baroclinic disturbance which results from the perturbation of an unstable jet. These integrations demonstrate the feasibility of using multiple, rotated, overlapping fine grids. Direct computations of the truncation error confirm the accuracy of the Richardson–type truncation error estimates.

The primitive equations do not form a well-posed Initial Boundary Value Problem (IBVP). The second adaptive model, based upon a non-hydrostatic set of equations which do form a well-posed IBVP, is developed and then tested by simulating a developing baroclinic disturbance. The well-posedness of the equations, the necessity for less filtering in the finite difference model and the ability to extend integrations to non-hydrostatic motions are significant reasons for using the new set of dynamical equations in place of the hydrostatic primitive equations.

Incorporating vertical refinement into an adaptive model is investigated. The ill-posedness of the primitive equations is a direct result of the hydrostatic approximation and may lead to instabilities in a vertically refining model. This is not a problem with the second set of equations. A more immediate and unsolved problem is that of vertically interpolating the thermodynamic variables of the hydrostatic approximation and the near geostrophic balance present in large scale flows. The atmosphere is very nearly in hydrostatic balance, thus even for the nonhydrostatic model the interpolation problem remains.

Acknowledgements

Life is Good

Foremost, I would like to thank Professors Joseph Oliger and Robert Street for their guidance and support throughout my stay at Stanford. They provided me with the opportunity to pursue new and exciting research in a field with which I was completely unacquainted, and for this I am forever in their debt. I would also like to thank my friends at Stanford, they have helped make this a pleasant experience.

My family, though far away, have provided much support and encouragement. Soon I hope to find the time to explain the them exactly what I've been doing.

I credit my continuing sanity to many friends outside the Stanford community, notably Eric Shrader, Pam Kincaid and my climbing partners (and most certainly good friends) John Burdekin, Jeff Kletsky and Tom Waschura.

My wonderful wife, of course. Without Anne life would not be nearly as good as it is. I thank her for her patience and understanding over the last three years. Words cannot express what needs to be acknowledged.

I would also like to thank the people at the Naval Enviromental Prediction Research Facility, particularly Drs. Tom Rosmond and Tim Hogan, for their comments about the work and their assistance using the machines at FNOC. I also thank my wife's parents for the superb accomodations and gracious hospitality I received from them during my trips to Monterey.

This work has been supported by the Office of Naval Research under contracts N00014-82-K-0035 and N00014-84-K-0267.

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By per Letter
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

Table of Contents

Abstract iv

Acknowledgements v

List of Figures viii

List of Tables x

1. Introduction 1

2. Review of the Adaptive Grid Refinement Technique 9

3. Primitive Equations 17
3.1 Equations 17
3.2 Discretization 20
3.3 Boundary Conditions 23
3.4 Primitive Equations Considerations 24

4. Test Case Results for the Primitive Equations 27
4.1 Barotropic Cyclone 27
4.2 Baroclinically Unstable Jet 33
4.3 Error Estimation 44

5. Vertical Refinement with the Primitive Equations 51
5.1 Interface Conditions 53
5.2 Initialization 63
5.3 Hydrostatic Balance 67
5.4 Failure of the Primitive Equations 71
5.5 Theoretical Considerations for Vertical Refinement with the Primitive Equations 72

6. Browning-Kreiss Equations 79
6.1 Derivation 80
6.2 Browning-Kreiss Solver 86
6.3 Vertical Refinement with the Browning-Kreiss Equations 89

7. Test Case Results For the Browning-Kreiss Equations 91
7.1 Baroclinically Unstable Jet 91
7.2 Error Estimation with the Browning-Kreiss Equations 103
7.3 Computing Non-hydrostatic Motions 104

8. Conclusions and Recommendations 105
8.1 Conclusions 105

8.2 Recommendations 107
Appendix 1 109
Appendix 2 119
Appendix 3 121
References 125

List of Figures

Figures

1 Dannenhoffer and Baron grid 3
2 Berger and Jameson grid 3
3 Grid setup for the NGM 5
4 Adaptive run with two levels of refinement 11
5 Tree for grids in Figure 4 15
6 Grid C 21
7 Vertical structure of the σ coordinate grid 21
8 Initial conditions for the barotropic cyclone 28
9 Cyclone after 3 days, coarse grid model 28
10 Cyclone after 3 days, adaptive model 28
11 Typical error estimate, u velocity 28
12 Fine grid surface pressure at 9 hours 30
13 Absolute vorticity ($\sigma = 0.5$) at $t = 0$ for the baroclinically unstable jet 35
14 Temperature ($\sigma = 0.9$) at $t = 0$ 35
15 Velocity vectors ($\sigma = 0.9$) at $t = 0$ 36
16 Surface pressure at $t = 0$ 36
17 Surface pressure at $t = 72$ hours 38
18 Absolute vorticity at $t = 72$ hours 39
19 Temperature ($\sigma = 0.9$) at $t = 72$ hours 40
20 Horizontally averaged kinetic energy 42
21 Richardson truncation error estimate 46
22 Direct truncation error estimate 46
23 The domain Ω 54
24 Control volume centered about the point $\pi_{i,j}$ 56
25 Coarse grid – Fine grid 56
26 Average surface pressure 60
27 Average kinetic energy 61
28A Geopotential using different vertical interpolation methods 69
28B Temperatures derived from Figure 28A 69
29 Vertical discretization for the Browning-Kreiss solver 88
30A Initial jet, u' velocity 94
30B Perturbation pressure ϕ' 95
30C s perturbation s' 96
31 Velocity vectors ($z = 1$km) at adaptive run start 97
32 Pressure ($z = 1$km) 97
33 Potential temperature ($z = 1$km) 98
34 Vertical cross section of v' 98
35 Pressure ($z = 1$km) after 36 hours 99

36 Potential temperature ($z = 1$km) after 36 hours 99
37 Fine grid positions 100
38 Pressure ($z = 1$km) on the fine grids 101
39 Potential temperature ($z = 1$km) on the fine grids 102

List of Tables

Table

1 Values of c_ν versus ν 75
2 c_ν versus ν with mean vertical wind shear 77
3 Standard atmosphere parameters for the Browning-Kreiss model 87

1. INTRODUCTION

Accurately computing atmospheric flows is a difficult task. Integrations must be performed accurately and in a timely manner if the simulations are to be helpful to weather forecasters, but the variety of scales which must be included for successful representation of the flow, and about which forecasters need information, strain the storage capabilities and speed of the most advanced computers. Planetary waves are many thousands of kilometers long and exist for weeks. At the opposite end of the scale of meteorological interest are the turbulent eddies which have length scales of a few centimeters and last only seconds. We are far from resolving all scales of motion in a single computation. We typically resolve only a small subset of the scales, parameterize others and disregard what is left. Unfortunately this is often not adequate.

In many cases the most interesting and important phenomena, from a forecasting perspective, are not properly resolved in global or regional scale weather models. These phenomena include structures such as tropical cyclones, strong surface fronts, rainbands, squall lines and jet streams. Research models have been constructed for use in studying these phenomena but general models for predictive purposes are not yet feasible. Present forecast models lack the resolution necessary to represent the small scale structure.

From a global perspective tropical cyclones, fronts, jet streams and other atmospheric phenomena are spatially and temporally localized. Difficulty in providing adequate resolution arises from our inability to know beforehand where these phenomena will occur. These are spatially and temporally local phenomena and local phenomena should be handled adaptively, but no adaptive atmospheric flow solvers exist.

Phenomena in many other fluid flows which are difficult to resolve are often localized. Adaptive solvers do exist for many of these flows and, in general, two adaptive strategies are used. In the first all existing gridpoints are redistributed from regions of small solution variation to regions of large solution variation. These global methods vary in the criteria and methods used to move the points, but in all cases the total number of points remains the same. They are often used in conjunction with grid transformation methods which involve mapping an irregular physical domain into

a rectangular computational domain. The second strategy involves adding or deleting grid points so as to obtain a desired solution accuracy. The additions and deletions are local, thus the techniques are *local* grid refinement techniques.

Atmospheric flows appear ideally suited to local grid refinement techniques because the important phenomena are localized. The local grid refinement techniques can be broken into two catagories: one in which the new points are inserted or imbedded into the existing grid, and hence only one grid exists, and a second where refinements are placed over the existing grid, the refinements constituting separate grids.

A example of embedding new points in an existing grid is the work of Dannenhoffer and Baron (1986). Their code solves transonic flow over a 2-D airfoil. Refinements are based on refinement parameters such as first or second order differences in the density, pressure or entropy. An expert system handles the refinement parameters and rules governing how and where to refine.

In this technique grids are no longer rectangular in nature and neither is the data structure which holds the solution fields. A significant amount of information must be stored to describe the grid structure. The solver which uses this grid structure is complex. An example of a refined grid is shown in Figure 1. Even with this complexity and loss of rectangularity much vectorization of the code is possible and efficient integrations are being performed.

An example where refinements are placed over the existing grid is the scheme of Berger and Oliger (1984). The same scheme has been used by Berger and Jameson (1985) who also solve transonic flow over a 2-D airfoil. In this technique the refinements are separate rectangular grids rather than being points embedded in the coarse grid. Any solver which works on a rectangle can be used, because the solver is just a module called by the adaptive routines to advance the solution on a single rectangular grid. Figure 2 shows a final set of grids in a Berger and Jameson calculation. Note the large difference between these grids and those of Dannenhoffer and Baron in Figure 1. There are many other differences between the two schemes and interested readers should consult the referenced papers.

There are no adaptive solvers for atmospheric flows but there are several approaches that are presently being used to address the resolution problem in numerical

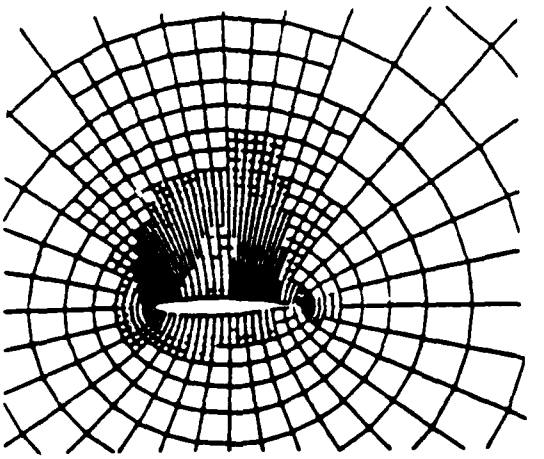


Figure 1 Grid after three regriddings in the Dannenhoffer and Baron grid refinement method. All points belong to the same grid.

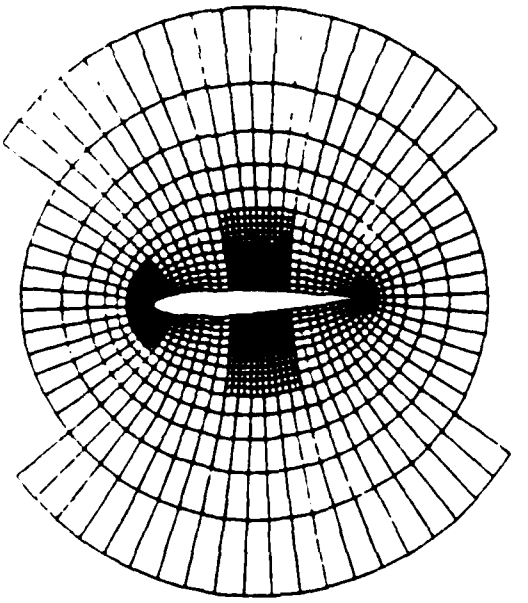


Figure 2 Grid after one refinement in the Berger and Jameson model. The fine grids are seperate from the coarse grid.

weather prediction (NWP). Grids are often vertically nonuniform with more layers close to the surface. Different vertical coordinates have been used to achieve this in a more natural way. For example, the potential temperature [$\theta = T(p_{surface}/p)^{R/c_p}$] may be used as the vertical coordinate with a resulting increase in resolution around fronts. The normalized pressure $\sigma = p/p_{surface}$ is more commonly used as a vertical coordinate. This system allows increased accuracy in the specification of the surface boundary because of the simplicity of the surface boundary condition. Horizontal layers smoothly follow the surface. These approaches are passive methods for increasing resolution. They are a direct result of the form of the equations or difference methods.

Another technique for increasing the resolution in atmospheric prediction codes consists of placing finer grids inside a coarser grid at any location where greater resolution or accuracy is desired. In the atmospheric science literature the fine grids are known as nested grids and implementation has been achieved in two forms: one-way interactive and two way interactive.

One way interaction is the simplest nested grid approach. The lateral boundary conditions for the fine grid are supplied from the coarse grid solution. Information is passed from the coarse grid to the fine grid but not from fine to coarse, hence, the method is called one-way interactive. Many regional atmospheric models are one-way interactive because they receive their lateral boundary values from global model solutions but have no effect on these solutions. The larger scales of the flow which cannot be simulated on the fine grid are allowed to affect the fine grid solution. The major problem is the different wave speeds which result from the different resolutions on the two grids. Discontinuities and distortions can develop at the fine grid boundary as a result of having no feedback from the fine to the coarse grid (Haltiner and Williams, 1980). Sponge-type boundary conditions have been developed for one-way interactive fine grid models which make useful results possible.

The one-way interactive approach implicitly assumes that small scale phenomena have no major influence on the larger scale flow treated on the coarse grid. This is not generally true because two-way exchanges of energy exist between scales. The two-way interactive nested grid approach addresses this problem. The procedure consists of integrating the fine grid along with the coarse grid. Lateral boundary conditions for the fine grid are taken from the coarse grid solution. The solution on the coarse

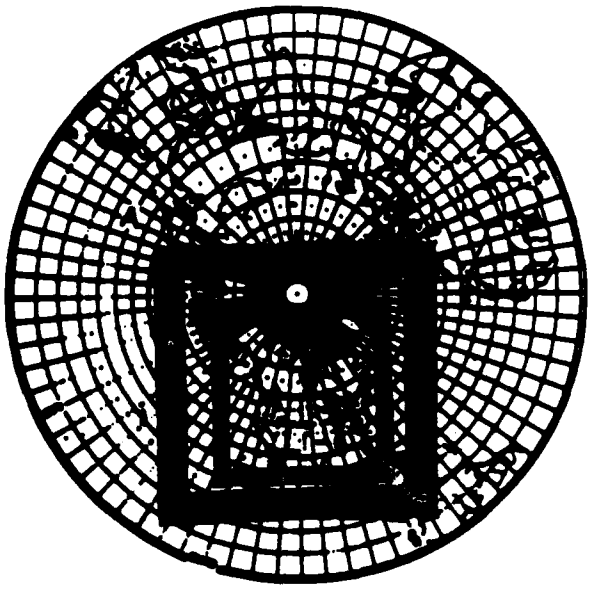


Figure 3 Grid setup for the NGM. Coarse global grid has an average resolution of 320 km., the second grid 160 km. and the third grid 80 km. From Hoke (1984).

grid is updated with the fine grid solution at any location where the two coincide. The scheme is thus two-way interactive because the fine grid solution does influence the coarse grid solution.

The two-way interactive method has been implemented in two forms, one where the fine grid location is fixed and another where the fine grid is allowed to move during the time integration. An example of the former is the Nested Grid Model (NGM) which has been developed at the National Meteorological Center. The National Weather Service distributes NGM results as guidance to forecasters. The model consists of a hemispherical grid over which two finer grids have been placed. The fine grids are centered over North America because forecast information is needed there. The grids are illustrated in Figure 3.

Several tropical cyclone models have fine grids which move during the integration to keep the fine grid over the cyclone. This is accomplished by moving the fine grid when a solution feature, such as the surface low associated with the cyclone, moves. In all cases the fine grids are aligned with the coarse grids, but they may move incrementally up, down or sideways. Examples of these models are the tropical cyclone models described by Harrison (1973) and Jones (1977).

It is difficult to know, a priori, precisely where increased resolution will be necessary. The NGM provides increased resolution over a continent because increased

resolution may be necessary there and also because forecasts are most important over the continent. The tropical cyclone models attempt to provide resolution of the cyclone and the the initial location of the cyclone must be known to locate the fine grid. The fine grid will remain even if the cyclone disappears, and if a new cyclone were to appear elsewhere no new fine grid would appear over it. The NGM is certainly not adaptive and the nested tropical cyclone models are not truly adaptive.

Solver complexity is a strong barrier to using adaptive techniques in NWP. Weather prediction codes solve much more than a simple set of dynamical equations. There are equations for water vapor (and possibly water in its other states), routines which calculate radiational heating and heating due to phase changes of water, routines which model cloud effects, complex calculations for fluxes of heat and moisture into the atmosphere, and usually parameterizations of other processes. Most codes are the result of many peoples' effort over several years. Proven and tested routines are often borrowed from one code to put into another. Adaptive methods using refinements which are separate grids appear the logical choice for use as the basis of an adaptive weather model. Existing, well-tested software can often be used with only minor changes and procedures can be written with little knowledge of the adaptive routines.

In this dissertation we present results from an adaptive atmospheric flow solver which uses the method and software developed by Berger and Oliger (1984). The adaptive method operates on multiple, component grids. Fine grids, which overlie the coarse grid(s), are created and removed based on a Richardson-type estimate of the truncation error in the finite difference solution. The goal is to maintain a given accuracy for a minimum amount of work.

Our purpose is to show that an adaptive atmospheric flow solver is feasible and that the adaptive technique will produce self-consistent results. In essence we are proving a concept, the concept being (1) that refinement should occur only where necessary, as dictated by the error in the numerical solution, and in this way improve the accuracy and overall resolution of the entire solution and (2) that this can be accomplished by using the method of Berger and Oliger. Hence, we wish to demonstrate that our adaptive model yields better results compared to results from the same solver on a single grid, this being sufficient to demonstrate the feasibility of the adaptive

atmospheric flow solver. For prediction purposes one would use the best available solver for the scales one is attempting to predict.

In Chapter 2 we review the adaptive grid refinement technique of Berger and Oliger. The solution procedure is outlined along with a description of the data structure and program design. Chapter 3 presents the hydrostatic "primitive equations" of meteorology which are derived from the Euler equations by making the hydrostatic approximation. A solver for these equations is used as the basis of our first adaptive solver. The initial test cases for this solver describe flow in an idealized atmosphere: adiabatic flow in a periodic channel with no moisture present in the atmosphere and no diabatic heating. At the end of the Chapter 3 we briefly discuss the issues of stability and accuracy for these equations as they are used in the adaptive model. More detailed discussion of the issues of accuracy and conservation is contained in Chapter 5. In Chapter 4 we examine the results of two simulations, one for a barotropic cyclone and another for a baroclinically unstable jet, and show that our adaptive model is self consistent and successful in simulating these flows. We also examine the truncation error estimates at the end of Chapter 4. Chapter 5 outlines the problems we have encoutered attempting to implement uniform vertical refinement into an adaptive or nested model. In the chapter we illustrate problems with the primitive equations. In Chapter 6 we present a new set of equations; the Browning-Kreiss (BK) equations. These equations no longer use the hydrostatic approximation, however, problems remain to be solved before a vertically refining model can be constructed with these equations. The BK equations offer several advantages over the primitive equations, most importantly their ability to be used for calculating non-hydrostatic motions. In Chapter 7 we present preliminary results from an adaptive solver based upon the Browning-Kreiss equations which indicate that the equations correctly represent large-scale atmospheric motions and are suitable for use in an adaptive model. The computation of non-hydrostatic motions with this model is also discussed. Conclusions and recommendations are found in Chapter 8.

2. REVIEW OF THE ADAPTIVE GRID REFINEMENT TECHNIQUE

We describe the adaptive procedure as used for 2-D hyperbolic problems. For large-scale atmospheric flows the horizontal scales are orders of magnitude larger than the vertical scales. Thus, we can easily treat atmospheric flow as a 2-D grid refinement problem even though it is a 3-D flow. We have constructed 2-D horizontally refining adaptive models and present results from these models in Chapters 4 and 7. We have not successfully constructed a model which refines in all three coordinate directions.

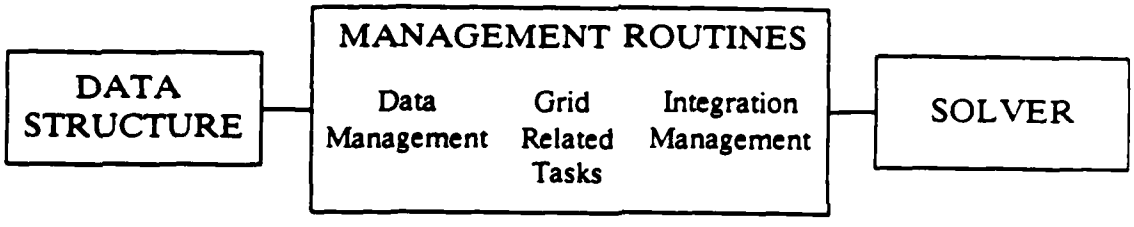
The adaptive method is based on the idea of using multiple, component grids on which the partial differential equations are solved. Refined grids are created or removed based on a Richardson-type estimate of the truncation error in the finite difference solution. The goal is to maintain a given accuracy in the numerical solution for a minimum amount of work. A complete description of the method can be found in Berger (1982) and Berger and Oliger (1984).

The solution procedure for the adaptive grid method is as follows. We begin with a solution on a coarse grid that has been integrated to some time t. The error introduced through the use of the numerical procedure is estimated at grid points and where these errors are judged to be too large the points are flagged. Then 2-D rectangular grids with finer meshes are fitted around these flagged points. These subgrids may have orientations differing from the coarse grid. Initial and boundary conditions for these new fine grids are taken from the coarse grid solution and the fine grids are integrated along with the coarse grid to a new time $t + \Delta t_c$, where Δt_c is the coarse grid time step. Smaller time steps are taken on the fine grids to keep $\Delta x / \Delta t$ constant. The coarse grid solution is then updated with the more accurate fine grid solutions. Errors may also be estimated on the fine grids and still finer grids introduced. Thus, there can be several levels of fine grids. The errors on the grids can be re-estimated every few time steps and then new fine grids can be created and old fine grids removed.

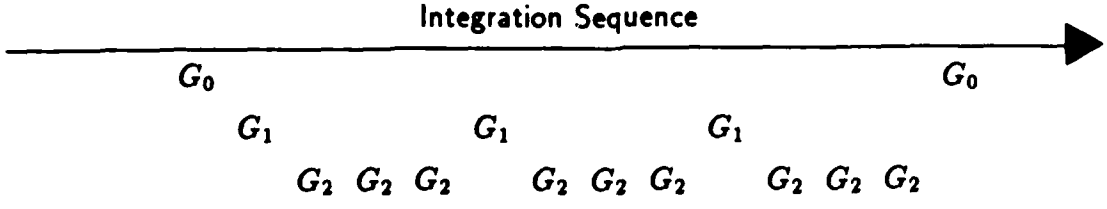
The large errors in the numerical solution are usually associated with sharp variations in the solutions, e.g., at fronts or other disturbances. By re-estimating the error after several time steps and regridding we create a scheme whereby the fine grids track the disturbances. The fine grids' arbitrary orientations allow them to align with

the disturbances which minimizes the size of the fine grids and provides for better resolution in numerical schemes.

The program can be viewed as consisting of three components; 1) a data structure, 2) a solver and 3) management routines. Due to the constructs of FORTRAN the data structure is fixed. It stores the solution vectors for all grids and information about these grids. The information and where it is stored are altered by the management routines. These routines also pass to the solver the locations of the solution vectors in the data structure. The overall program can be viewed as the interaction between the solver and data structure with the management routines controlling this interaction and performing the necessary intergrid communication (setting boundary conditions and updating).



The key algorithms are those which perform the integration, error estimation and grid generation. To illustrate how they work, consider the grid arrangement shown in Figure 4. There are several ways to advance the solution by one Δt_c on the grids. These are dependent on the interface conditions between the grids. For example, with a refinement ratio $r = 3$, ($r = h_c/h_f$ = the ratio of the coarse grid Δx to the fine grid Δx) the integration order (from coarsest to finest and left to right) is



where G_i are grids at level i with i increasing for finer levels. Grids at level i are integrated r times as often as grids at level $i-1$ but with $\Delta t_i = \Delta t_{i-1}/r$, thus, all are integrated to the same point in time. The order of integration assures that all fine grids will have sources for boundary values.

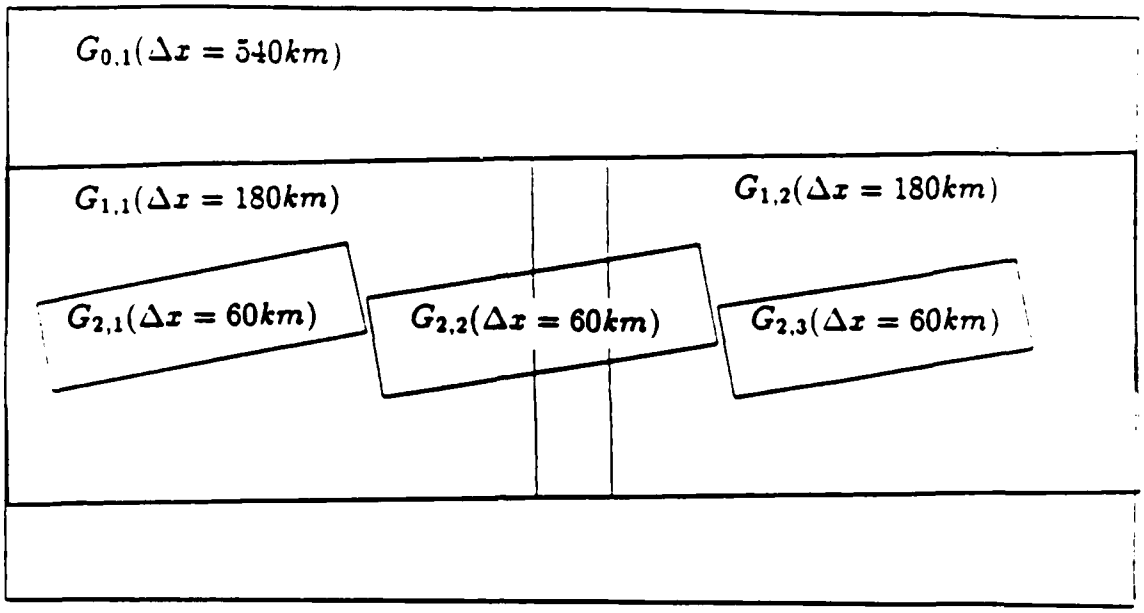


Figure 4 Adaptive run with two levels of refinement. $G_{0,1}$ is the coarse grid, $G_{1,i}$ fine and $G_{2,i}$ finer resolution grids.

Error estimation is also repeated and solutions from the fine grids must be passed to the coarse grids. Errors are estimated and grids replaced on each level after a number of time steps specified by the user. Grids at level i will be replaced, based on an error estimate on level $i-1$ grids, r times more often than grids at level $i-1$. Solutions on level i are updated with those on a higher level when the higher level solutions have reached the same point in time.

Errors on the various grids are estimated using a method based on Richardson extrapolation. If the solution is smooth the local truncation error can be expressed as

$$\begin{aligned} u(x,t+k) - Q_h(u(x,t)) =& k(k^{q_1}a(x,t) + h^{q_2}b(x,t)) \\ &+ k\,O(k^{q_1+1} + h^{q_2+1}) \\ =& \tau + k\,O(k^{q_1+1} + h^{q_2+1}), \end{aligned} \tag{2.1}$$

where q_1, q_2 are the orders of accuracy in time and space, h and k are the step sizes in space and time respectively and Q_h is an operator representing the finite difference scheme and defined as $\tilde{u}(x,t+k) = Q_h\big(u(x,t)\big)$, where $\tilde{u}$ is the approximate solution. Taking two time steps with the method defined by Q results in a truncation error of

$$u(x,t+2k) - Q_h^2(u(x,t)) = 2\tau + kO(k^{q_1+1} + h^{q_2+1})$$

where Q_h^2 is the operator Q_h twice applied to $u(x,t)$. If we let Q_{2h} be the same difference operator with step sizes of $2k$ and $2h$ and if we assume that the order of accuracy of the time and space differencing are equal and also that the solution is sufficiently smooth, then the expression for the truncation error associated with this operator is

$$u(x,t+2k) - Q_{2h}(u(x,t)) = 2^{q+1}\tau + kO(k^{q_1+1} + h^{q_2+1})$$

and the expression for the leading order term τ of the truncation error is

$$\frac{Q_h^2(u(x,t)) - Q_{2h}(u(x,t))}{2^{q+1}-2} = \tau + O(h^{q+2}). \tag{2.2}$$

This gives us an estimate of the local truncation error at time t. The procedure is to take a giant step based on mesh widths $2h$ and $2k$ using the solution at time t and then to compare it with the solution found by taking two regular steps.

Several features of this method favor its use. First, the exact form of the truncation error need not be known because the functions $a(x,t)$ and $b(x,t)$ of (2.1) are never calculated. Second, for systems of equations containing several variables determining the truncation error and calculating it accurately can be very difficult. Here the same solver used to integrate the equations can be used to estimate the error. The estimator is independent of the finite difference method and is also indepedent of the PDE. Finally, although the method does not produce accurate estimations of τ for nonsmooth solutions, qualitatively appropriate results are obtained because the estimates will be large in these regions and will lead to the desired regridding.

This procedure is used to calculate the truncation error, as defined by equation 2.1, for hyperbolic partial differential equations. For elliptic PDE's, Caruso (1985) uses a similar procedure to calculate the *solution* error and then uses the solution error to calculate the truncation error. The truncation error and the solution error are generally not the same and further discussion of the truncation and solution errors can be found in Caruso (1985).

Estimating the error at grid points is the first step in the regridding procedure. The regridding procedure is

1) flag points needing refinement,

2) cluster the flagged points,

3) fit rectangular grids around the clustered points and

4) repeat step 2 and 3, using different methods, if necessary.

Gridpoints are flagged if the estimated error exceeds a value specified by the user. Clustering the flagged points serves two purposes. First, it separates spatially distinct phenomena. In many problems there are often multiple shocks or fronts. These features will then be on different grids. The second purpose is to subdivide points when one large region should be fit with several grids.

Clustering the grid points and fitting rectangles to the clusters are the most difficult parts of the regridding task. Inexpensive clustering algorithms are rarely satisfactory for both clustering purposes. For this reason a simple algorithm is used for clustering in a first pass and, if this proves unsatisfactory, a more complex and expensive algorithm is used. The simple method produces clusters according to the nearest neighbor principle. If a point has any other point within some specified minimum distance then these points are in the same cluster. This method works well for the first purpose, but very poorly for the second. The more complex methods use minimum spanning trees or nearest neighbor graphs. These structures connect the points in an organized way. An iterative method is then used which merges points with core groups of points and attempts to maximize the efficiency of a grid. The efficiency of a grid is a measure of how large the grid is compared to how many flagged points the grid contains.

The last task the regridding algorithm must complete is to fit the rectangles to the clusters. There are several methods which will perform this task and some produce, on the average, more efficient rectangles than others. In the present algorithm a simple, inexpensive method is used because it works well and also must frequently be used in the clustering routine. The method fits a rectangle by computing a least-squares fit line to a given cluster of points. This line is the principal axis of the rectangle and an orthogonal line will be parallel to the other axis. It is then an easy matter to compute where the sides and the corners of the rectangles should be, though this is the most expensive part of fitting the rectangle. Finally, the rectangle is enlarged so as to provide a buffer zone between the flagged points (the phenonmena) and the rectangle (fine grid) boundaries.

The data structure stores two kinds of information – descriptions of grids and

the grid solution vectors. It is natural to think of these grids in the context of a tree, with the coarse grid being at the root of the tree. At each node of the tree lies a grid. Each grid (node) is characterized by the following:

1) grid location,
2) grid point specifications,
3) level in tree,
4) offspring pointer,
5) sibling pointer,
6) parent pointer,
7) intersection pointer,
8) pointer to the next grid at same level,
9) time to which grid has been integrated,
10) index in storage array for solution on grid.

This information is stored for each grid at the nodes of the tree. Figure 5 shows an example of the tree for the grid system of Figure 4.

All solution vectors are stored in one array. This array is managed as a linked list of used and available blocks of storage. Storage is allocated in contiguous blocks by scanning the list of available blocks, taking the first block that is large enough, and returning whatever space is unused. Reclaimed space can be re-inserted into the list and reused. The structure of FORTRAN does not allow for dynamic memory allocation outside the program, thus, all storage is defined initially.

We have described the algorithms which control the integration sequence, error estimation, clustering, gridfitting and data management. The program is constructed modularly. The data structure and the methods used to alter it can be accessed by all routines. The user has only to supply an integration routine (a solver) which solves the equations which he is interested in. Changes usually require altering only one or a few modules, and not the entire code.

Our use of the adaptive grid method and the program just described is greatly facilitated by code modularity. The program contains the necessary algorithms and data structures to carry out the adaptive method outlined earlier. Many of these algorithms have been borrowed from computer science, mathematics and other disciplines and it is their application to numerical weather prediction that is new.

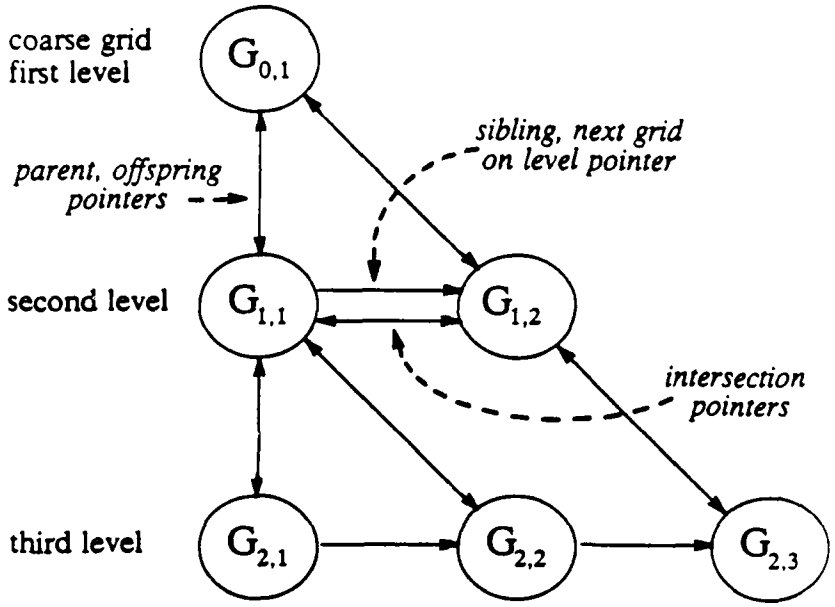


Figure 5 Tree for grids in Figure 4. Included are some of the pointers listed in the text.

3. PRIMITIVE EQUATIONS

3.1 Equations

We solve the Euler equations for dry, adiabatic, large-scale atmospheric flows. The set commonly used for studying compressible fluid flows has as dependent variables u, v and w, the horizontal and vertical velocities, and T, p and ρ, the temperature, pressure and density, with geometric coordinates x, y and z. We make several changes to this set.

Large-scale flows are very nearly hydrostatic, and making this approximation reduces the z momentum equation to the hydrostatic equation with the added benefit of removing sound waves from the solution. We also recast the system by using the nondimensional pressure σ in place the vertical coordinate z. The coordinate σ is defined as

$$\sigma = p/\pi$$

where π is the surface pressure. Thus, at the surface $p = p_s = \pi$, $\sigma = 1$ and at the top of the atmosphere $p = 0$ and $\sigma = 0$. The vertical coordinate σ has a range $0 \leq \sigma \leq 1$.

Using σ, we can write the equation of state as $\sigma\pi = \rho RT$ and use it to eliminate the density ρ from the equations. Finally, we introduce the geopotential $\phi = gz$, which replaces z as a dependent variable. The reduced Euler equations, known as the hydrostatic primitive equations, are

$$\frac{\partial}{\partial t}(\pi u) = -\frac{\partial}{\partial x}(\pi u^2) - \frac{\partial}{\partial y}(\pi uv) - \frac{\partial}{\partial \sigma}(\pi u\dot{\sigma}) + \pi f v - \pi\frac{\partial \phi}{\partial x} - RT\frac{\partial \pi}{\partial x} + \pi F_x \tag{3.1}$$

$$\frac{\partial}{\partial t}(\pi v) = -\frac{\partial}{\partial x}(\pi vu) - \frac{\partial}{\partial y}(\pi v^2) - \frac{\partial}{\partial \sigma}(\pi v\dot{\sigma}) - \pi f u - \pi\frac{\partial \phi}{\partial y} - RT\frac{\partial \pi}{\partial y} + \pi F_y \tag{3.2}$$

$$\frac{\partial \phi}{\partial(ln\,\sigma)} = -RT \tag{3.3}$$

$$\frac{\partial}{\partial t}(\pi c_p T) = -\nabla_\sigma \cdot (\pi c_p T V) - \frac{\partial}{\partial \sigma}(\pi c_p T\dot{\sigma}) + \pi Q + RT\omega + \pi F_T \tag{3.4}$$

$$\frac{\partial \pi}{\partial t} = -\nabla_\sigma \cdot \pi V - \pi\frac{\partial \dot{\sigma}}{\partial \sigma} + F_\pi \tag{3.5}$$

where

$$V = u\vec{i} + v\vec{j}$$
$$R = \text{gas constant}$$
$$\dot{\sigma} = \frac{\partial \sigma}{\partial t}$$
$$\omega = dp/dt = \pi\dot{\sigma} + \sigma(\partial\pi/\partial t + V \cdot \nabla_\sigma \pi).$$

The independent variables are x, y, σ and t and the dependent variables are u,v,$\dot{\sigma}$,ϕ,π and T. Equations 3.1 and 3.2 are the u and v momentum equations, 3.3 is the first law of thermodynamics and 3.4 the hydrostatic equation. Equation 3.5 is the transformed continuity equation. A complete derivation of these equations can be found in Holton (1979) and Haltiner and Williams (1980).

The need to make several other assumptions arises when these equations are used. The terms F_x,F_y, and Q are forcing or source terms that account for processes not explicitly accounted for in the dynamics equations. The terms in the momentum equations theoretically include the effects of diffusion, both turbulent and molecular. In the thermodynamic equation Q represents latent and radiational heating and cooling, fluxes of heat from the boundaries and, in essence, all diabatic effects. Major assumptions underlying models often are found in the parameterizations of these terms.

In large-scale flows the effects of viscosity and turbulence have negligible contributions to the forcing terms F_x and F_y. Horizontal diffusion, fourth order in the interior and second order near the grid boundaries, is included only to stabilize the numerical solution. This stabilizing diffusion term is also calculated for F_π and F_T. There is no vertical diffusion in any of the equations. We are solving for adiabatic flow, the model does not contain radiation effects, and there are no sources of heat.

Temperature instabilities are usually addressed using convective adjustment parameterizations. Unstably stratified air is seldom observed in the atmosphere at large-scales because convection (vertical air motion) takes place in the atmosphere as a response to the instability, leaving the atmosphere stably stratified. The instability can be thought of as the presence of more dense air above less dense air. The use of the hydrostatic assumption in the equations removes the mechanism which allows the atmosphere to respond to instabilities in a vertical air column. The instability

cannot give rise to vertical motions in these equations because there is no feedback to a vertical momentum equation in which $\partial w/\partial t$ is driven by the forcing. A common approach is to parameterize convection that arises from these instabilities through the use of convective adjustment schemes. We describe a simple dry convective adjustment scheme used in this solver. For a more detailed explanation see Haltiner and Williams (1980).

Air parcels can be characterized by their lapse rates γ,

$$\gamma = -\frac{\delta T}{\delta z}.$$

If in a dry atmosphere the lapse rate at a point is greater than the dry adiabatic lapse rate, then a vertical adiabatic displacement of a parcel at this point will be unstable. This will produce vertical convection in the region. A simple way to parameterize this process is as follows:

1) Calculate the large-scale fields without considering instabilities.
2) Calculate the actual lapse rates and dry adiabatic lapse rates in a column.
3) Compare the lapse rates

$\gamma \leq \gamma_d$	*stable*	$\delta T = 0$
$\gamma > \gamma_d$	*unstable*	$\delta T \neq 0$

where

γ_d is the dry adiabatic lapse rate and
δT is temperature change in the column due to convection.

In the first case nothing is done because the column is stable. In the second case the vertical temperature profile is adjusted to a neutral or slightly stable lapse rate γ_d subject to the condition that total potential energy is conserved, i.e.,

$$\int_{Zb}^{Zt} c_p \delta T \rho dz = \frac{c_p}{g} \int_{p_t}^{p_b} \delta T dp = 0$$

where b and t represent the bottom and top of the unstable layer and c_p is the specific heat of air at constant pressure. This scheme assumes that convection causes potential energy to be converted to kinetic energy which is eventually converted into internal energy.

3.2 Discretization

On a horizontal plane the grid used in this model is the C grid described in Arakawa and Lamb (1977). The C grid is shown in Figure 6. It resolves shorter waves well, accurately represents wave group velocities and amplitudes, and possesses very good geostrophic adjustment properties. Vertical differencing takes place on the grid illustrated in Figure 7. The variables shown on the horizontal grid are carried between σ levels and σ and its time derivative are carried at σ levels. The surface pressure π and the surface geopotential ϕ_s are defined at the surface. The σ axis is always vertical, i.e, radially outward from the center of the earth. Consequently, the values or gradients of π or ϕ_s needed for a computation at some point on a horizontal plane above the surface are taken as those values at the surface directly (vertically) below the point at the surface.

We first consider how the equations are differenced on the horizontal σ surfaces on the C grid. By centering a control volume over a u velocity point at (x, y), where $x = i \cdot \Delta x$ and $y = j \cdot \Delta y$, we can denote fluxes of u–momentum through the control volume faces in the x and y direction as F^u and G^u. The discretization for the horizontal advection terms in the u–momentum equation is

$$\frac{\partial}{\partial x}(\pi u u) + \frac{\partial}{\partial y}(\pi u v) = \frac{1}{\Delta x}(F^u_{i+\frac{1}{2},j} - F^u_{i-\frac{1}{2},j}) + \frac{1}{\Delta y}(G^u_{i,j+\frac{1}{2}} - G^u_{i,j-\frac{1}{2}})$$

where

$$\begin{aligned} F^u_{i+\frac{1}{2},j} &= \frac{1}{8}(u_{i+1,j} + u_{i,j}) \\ &\quad \cdot \left(u_{i,j}(\pi_{i-\frac{1}{2},j} + \pi_{i+\frac{1}{2},j}) + u_{i+1,j}(\pi_{i+\frac{1}{2},j} + \pi_{i+\frac{3}{2},j})\right) \\ G^u_{i,j+\frac{1}{2}} &= \frac{1}{8}(u_{i,j+1} + u_{i,j}) \\ &\quad \cdot \left(v_{i+\frac{1}{2},j+\frac{1}{2}}(\pi_{i+\frac{1}{2},j} + \pi_{i+\frac{1}{2},j+1}) + v_{i-\frac{1}{2},j+\frac{1}{2}}(\pi_{i-\frac{1}{2},j} + \pi_{i-\frac{1}{2},j+1})\right). \end{aligned}$$

Vertical (σ) derivatives, for example $(\partial(\pi u \dot{\sigma})/\partial\sigma)$, are computed as

$$\frac{\partial}{\partial \sigma}(\pi u \dot{\sigma}) = \frac{\overline{\pi_{i,j}}}{2\Delta\sigma_k}\left(\overline{\dot{\sigma}_{i,j,k+\frac{1}{2}}}(u_{i,j,k+1} + u_{i,j,k}) - \overline{\dot{\sigma}_{i,j,k-\frac{1}{2}}}(u_{i,j,k} + u_{i,j,k-1})\right),$$

where the overbar denotes an averaged value for π and $\dot{\sigma}$ at the points (i,j). The velocity u is averaged to compute an approximation of u at $k \pm \frac{1}{2}$.

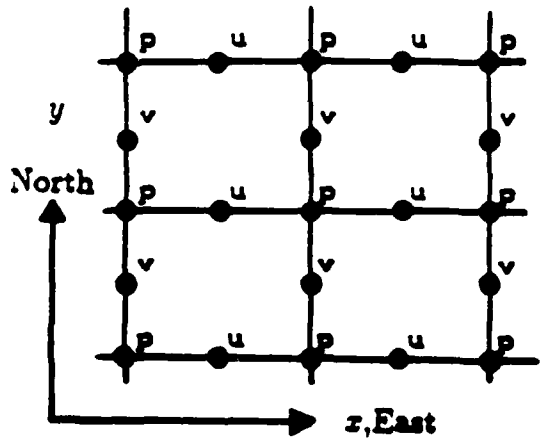


Figure 6 Grid C. The variables T, ϕ and π are found at the p points. u and v are the velocities in the x and y direction (East and North) respectively.

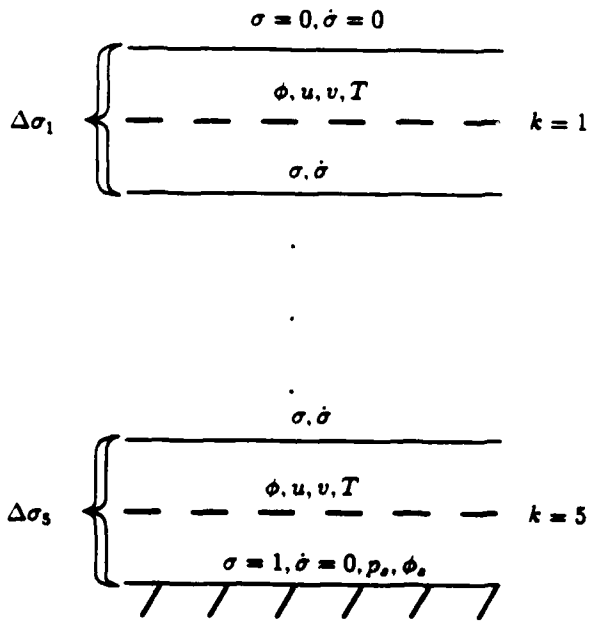


Figure 7 Vertical structure of the finite difference grid in the sigma (σ) coordinate system for a five level model.

Similarily, we can center a control volume about a π point and denote "mass" fluxes through the surfaces as F, G and $\dot{S}$. The continuity equation (3.5) is differenced as

$$\frac{\partial}{\partial t}\pi_{i,j} + \frac{1}{\Delta x}(F_{i+\frac{1}{2},j,k} - F_{i-\frac{1}{2},j,k}) + \frac{1}{\Delta y}(G_{i,j+\frac{1}{2},k} - G_{i+j-\frac{1}{2},k}) + \frac{1}{\Delta \sigma_k}(\dot{S}_{i,j,k+\frac{1}{2}} - \dot{S}_{i,j,k-\frac{1}{2}}) = 0$$

where

$$\dot{S}_{i,j,k+\frac{1}{2}} = \pi_{i,j}\dot{\sigma}_{i,j,k+\frac{1}{2}}$$
$$F_{i+\frac{1}{2},j,k} = \frac{1}{2}u_{i+\frac{1}{2},j,k}(\pi_{i,j} + \pi_{i+1,j})$$
$$G_{i,j+\frac{1}{2},k} = \frac{1}{2}v_{i,j+\frac{1}{2},k}(\pi_{i,j+1} + \pi_{i,j}).$$

The remaining terms in the equations 3.1–3.4 are differenced in a similar manner. The overall scheme will conserve mass (disregarding the diffusion term in the pressure tendency equation) but will not necessarily conserve energy.

The leapfrog method is used to integrate the spatially differenced equations in time. The method is explicit, second order in time and possesses good phase and amplitude characteristics for propagating waves. Equations 3.1, 3.2 and 3.3 are marched forward in time. The surface pressure π is found by integrating the continuity equation (3.5) vertically at each π gridpoint. It is only when integrating this equation that the vertical boundary conditions ($\dot{\sigma} = 0$ at $\sigma = 0, 1$) are needed. The vertical integration of 3.5 results in

$$\frac{\partial \pi}{\partial t} = \sum_{k=1}^{K} \nabla_\sigma \cdot (\pi V)\Delta\sigma_k.$$

The $\dot{\sigma}$'s can be found by integrating 3.5 down to the required level. The geopotential ϕ is found by integrating the hydrostatic equation 3.4 .

The stability of the leapfrog scheme is limited by the CFL condition

$$\frac{c\Delta t}{\Delta x} \leq \frac{1}{\sqrt{2}}$$

The fastest waves in these equations are the gravity waves, and the external gravity wave travels approximately an order of magnitude faster than the meteorological waves of interest. These waves are important in the geostrophic adjustment process; thus they cannot be filtered out of the equations without some adverse affects. However, the gravity waves severely limit the maximum time step which can be used.

3.3 Boundary Conditions

This problem is posed as an initial boundary value problem (IBVP). The initial values for the velocities u and v, the surface pressure p_s or π and the temperature T or the geopotential ϕ are necessary as initial conditions for the model.

Boundary conditions must be specified in the vertical at the top and bottom sigma layers and at all lateral boundaries. The choice of the sigma coordinates leaves us with very simple boundary conditions in the vertical. The conditions are

$$\dot{\sigma} = \frac{d\sigma}{dt} = 0$$

at both the top ($p = 0$, $\sigma = 0$) and the bottom ($p = p_s = \pi$, $\sigma = 1$) of the computational domain. At the surface ϕ_s is specified and this serves as the lower boundary condition for the integration of the hydrostatic equation 3.4 . At the upper and lower boundaries free slip conditions are applied for the u and v velocities and no-flux conditions are applied for the temperature.

The lateral boundary conditions may vary with the application of the model. Our test cases are for flow in a free slip, east-west periodic channel. The north–south boundaries are no flux. ($v = 0$). We use these boundary conditions on the base (coarse) grid.

For fine grid boundary conditions we specify u, v, T and π at the boundaries using bilinearly interpolated values from another grid. In this regard, we are choosing to apply continuity conditions at the fine grid boundaries as opposed to treating each fine grid as a separate initial boundary value problem. We call these boundary conditions continuity conditions because they specify that the fine and coarse grid solutions agree at the boundaries. We can do this because the fine–coarse grid boundaries are in regions of low solution error, thus, the coarse grid solution is accurate and the bilinear interpolation of the fine grid boundary values from the coarse grid introduces only small error. Conditions appropriate for initial boundary value problems, open boundary conditions in the case of the fine grids, can be derived by examining characteristics of the solution at the fine grid boundaries and specifying conditions such that the solution is uniquely determined, yet not overspecified. This procedure, in the case of the hydrostatic primitive equations, is discussed in Chapter 5, Section 5.

3.4 Primitive Equations Considerations

In this section we mention some of the theoretical and practical results which are relevent to adaptive calculations with the primitive equations.

Oliger and Sundström (1978) have shown that the primitive equations are ill-posed for the initial boundary value problem with open boundaries and local, pointwise boundary conditions. The question arises as to whether the primitive equations are ill-posed for solution on the local refinements (fine grids). This would be the case if we chose to treat a fine grid as a separate IBVP and use boundary conditions appropriate for the equations. Instead we have chosen to use continuity conditions and interpolate all data from one grid on to the boundary of another. The ill-posedness is not an issue if the continuity conditions are applied correctly, i.e., with sufficient accuracy. The ill-posedness of the equations are discussed in more detail in Chapter 5.

Limited area modellers have traditionally circumvented the ill-posedness problem and its resulting exponential error growth by including horizontal dissipation in their models and, more importantly, by using sponge-type boundary conditions and increased horizontal dissipation near lateral boundaries. This leaves open the question of exactly what equations are being solved and the accuracy of the limited area model solutions. Our calculations are for flow in a periodic channel. The primitive equations are weakly well-posed for these boundary conditions. Now, we must consider the accuracy and stability of these conditions.

There are no analyses for the primitive equations or for nonlinear hyperbolic equations concerning the accuracy or stability of our boundary scheme. For a 2-D model hyperbolic equation Berger (1985) has shown that using leapfrog with overlapping grids and grid refinement in both time and space is stable. Our adaptive integrations of the primitive equations have also proven to be stable experimentally.

Accuracy and conservation are related issues and there are few results concerning overlapping boundaries which are rotated with respect to each other. Berger (1984) derives a conservative boundary scheme for use in solving hyperbolic systems of conservation laws on 2-D rotated rectangles. We do not implement that scheme here and know of no implementation of it to date. Henshaw (1985) has found that when solving elliptic equations on overlapping grids one must use boundary-value interpolation schemes that are at least as accurate as the interior numerical scheme

and in some cases at least of one order higher accuracy in order to maintain the overall accuracy of the numerical scheme. Browning, Kreiss and Oliger (1973) show that solutions of hyperbolic equations on embedded grids of different resolutions may produce phenomena similar to that of the propagation of waves through materials of different densities. There can be interference of waves which have passed through refined regions with waves that have not. This interference is a weak instability and obviously results in extremely inaccurate solutions.

We address the issues of accuracy and conservation in how and where we decide to place fine grids. In the adaptive scheme, fine grid boundary values are interpolated bilinearly from other grids. We use an estimate of the error in the solution to place the fine grids and periodically re-estimate the error and replace the fine grids so that regions of high error always remain contained in the fine grids. The regions of high error (the disturbances) must never be allowed to pass through fine grid boundaries onto the coarse grid. Thus, fine grid boundaries always lie in areas where the solution error is low, i.e., where bilinear interpolation will introduce only small errors. Bilinear interpolation of the boundary values also insures a nearly conservative scheme whereas higher order interpolation results in significantly greater conservation errors. Our integrations indicate that, in this context, the use of bilinear interpolation is sufficient to insure accurate and stable solutions.

4. TEST CASE RESULTS FOR THE PRIMITIVE EQUATIONS

In this chapter we present results for two flow simulations using the adaptive primitive equations model. We present no detailed data concerning run times for different models. Our calculations indicate a breakeven point for using the adaptive method over a single fine grid at about 50-60% fine grid coverage in a 2 level adaptive model run. This is a research code and with some optimization the breakeven point could probably be improved to 70-80%.

4.1 Barotropic Cyclone

The first test case for the adaptive solver is the simulation of a barotropic cyclone (no vertical variation) which is being advected by an easterly flow in an east-west periodic channel. The primitive equations are solved on an f-plane (f = constant $= 5.0 \times 10^{-4} s^{-1}$) for this flow. The flow is initially barotropic and remains barotropic. There is no surface friction or energy sink other than the diffusion used to stabilize the computations, hence the solution should show the cyclone being advected towards the west with little change in its appearance.

The initial conditions are shown in Figure 8. The wind field is constructed by superposing a cyclone onto an easterly flow. The easterly flow has the form

$$u(y) = U_o \cdot \sin^2\left(\frac{\pi y}{L_y}\right)$$

where L_y is the width of the channel and $0 \leq y \leq L_y$. The symmetric cyclonic wind field superposed over the zonal flow has the form

$$U_T(x,y) = U_c\sqrt{\frac{(x-x_o)^2+(y-y_o)^2}{L_c^2}} \cdot \exp\left[\frac{1}{2}(1-\frac{(x-x_o)^2+(y-y_o)^2}{L_c^2})\right],$$

where U_T is the tangential wind velocity of a cyclone centered at (x_o, y_o). The results which follow are for the case $U_c = 20m/s$, $U_o = -10m/s$ and $L_c = 350km$.

The shallow water equations are sufficient to simulate a barotropic flow. We employ the full primitive equations, but by not allowing any vertical variation we are effectively solving the shallow water equations. The surface pressure π takes the place of the free surface height h in the shallow water equations when using the primitive

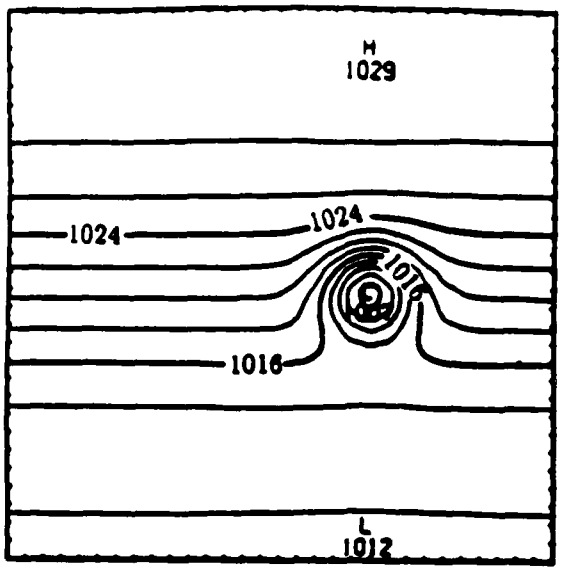


Figure 8 Initial conditions for the barotropic cyclone. Surface pressure is in millibars. Winds travel counterclockwise around the low pressure. The cyclone is being advected towards the left by a zonal wind.

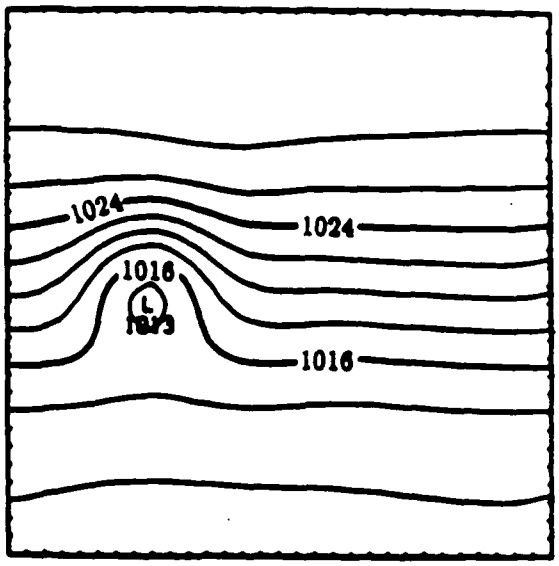


Figure 9 Cyclone after 3 days with the integration carried out on the coarse grid with $\Delta x = 180km$.

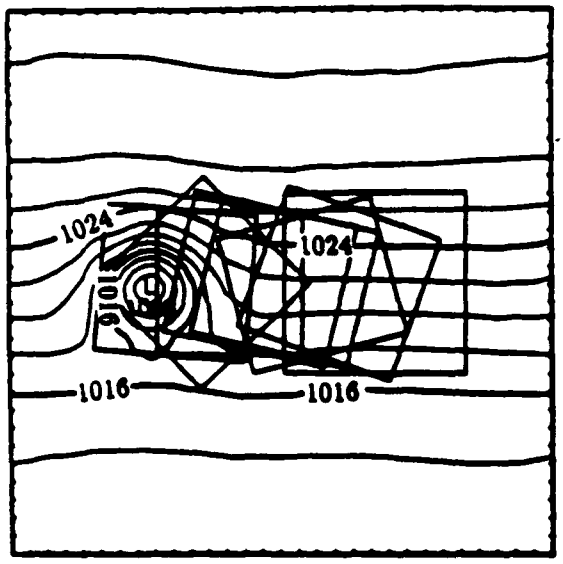


Figure 10 Cyclone after 3 days using the adaptive model. There is only a single fine grid in the region at any time. $\Delta x_{coarse} = 180km.$, $\Delta x_{fine} = 60km$. Fine grids used in the calculations are displayed.

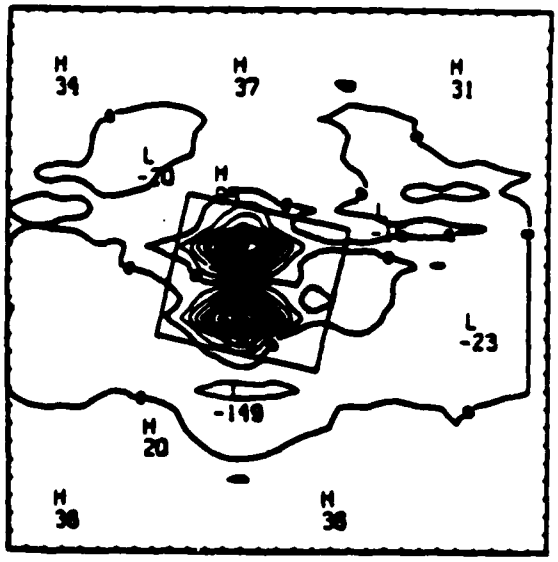


Figure 11 Typical error estimate for the u velocity (dimensionless $\times 10^4$). The estimate has been normalized by $|U|_{max} = 27\,m/s$ and the value $\tau = 0.035$ is used to flag points for regridding.

equations to simulate a barotropic flow. The surface pressure field depicted in Figure 8 is found by solving a Poisson equation which is derived by taking the divergence of the momentum equation and assuming that the divergence of the horizontal velocity field is zero. The Poisson equation is

$$\nabla^2\pi = -\frac{1}{T_{mean}}\left(\frac{\partial}{\partial x}\left(\frac{\partial(uu)}{\partial x} + \frac{\partial(uv)}{\partial y} - fv\right) + \frac{\partial}{\partial y}\left(\frac{\partial(uv)}{\partial x} + \frac{\partial(uv)}{\partial y} + fu\right)\right)$$

where T_{mean} is the horizontal mean of the temperature.

Figure 9 shows the solution after 3 days for a coarse grid run with $\Delta x = \Delta y =$ 180km. Figure 10 depicts an adaptive solution with a refinement ratio of three and one level of refinement. In the adaptive run the coarse grid size is $\Delta x = \Delta y = 180$km. and the fine grid size is $\Delta x = \Delta y = 60$km. These results clearly show that the coarse grid does not sufficiently resolve the cyclone resulting in the cyclone's premature decay while, in the adaptive run, the cyclone is resolved and shows very little decay. The fine grid is needed and the adaptive calculation is successful.

Figure 10 also shows the placement of the fine grids in the channel. The error estimation and regridding was performed every nine hours. There is only one fine grid in the channel at any one time and the plots show *all* the grids that are placed. The fine grids are usually just slightly larger than the cyclone and the regridding occurs often enough so that the fine grid tracks the cyclone. Also note that the fine grids in this simulation are not aligned with the coarse grid. Another simulation was performed where the fine grids are aligned and have points coincident with the coarse grid. The simulations differ only slightly and demonstrate that the orientation of the fine grids has little effect on the solution though it can have a large effect on the number of points required in a refinement large enough to cover regions of large error.

Fine grid placement depends on an error estimate of the coarse grid solution as described in Chapter 2. Only the velocity error estimates were used here to place fine grids. We will discuss error estimates for the surface pressure and temperature for the baroclinically unstable jet but they were not used for fine grid placement. Figure 11 shows a typical error estimate for the u velocity field along with the fine grid placed over a set of flagged points derived from the error estimate. For this flow the error estimate of the velocity fields places the fine grid over the cyclone - which is where we expect it would be necessary.

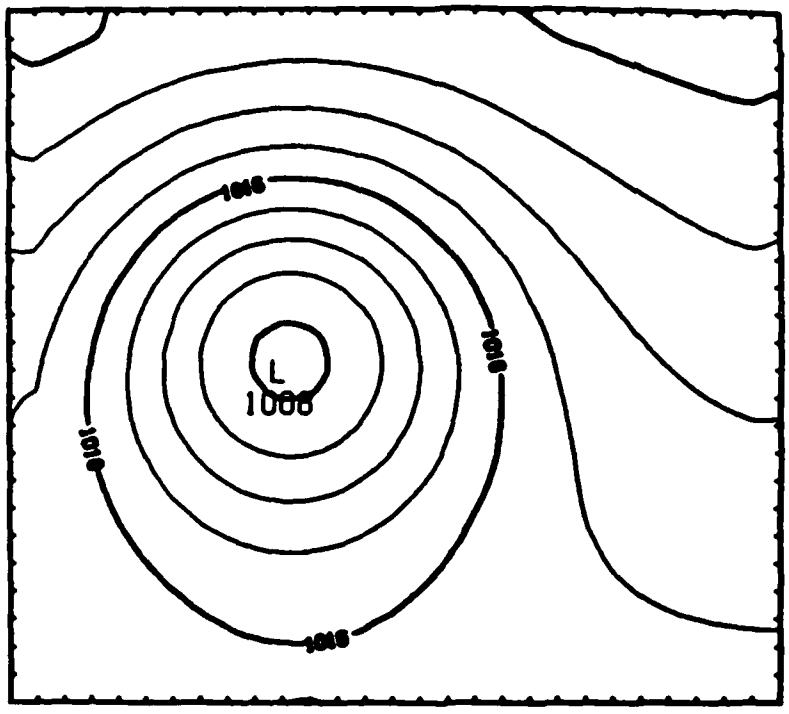


Figure 12 Surface pressure (mb) on a fine grid after 9 hours integration. Note the kinks at the boundaries. These kinks do not appear to affect the solution.

Initial conditions for fine grids are interpolated off the coarse grid or fine grids which existed before the regridding. A bicubic interpolation scheme is used to obtain these initial values. Bilinear interpolation has been tried, but was found to excite spurious gravity waves (noise) which take several hours to decay. This is not unexpected. Bilinear interpolation yields continuous functions but only piecewise continuous first derivatives between interpolation points. The discontinuities excite gravity waves, especially the discontinuities in the derivatives of the surface pressure and geopotential – derivatives which are crucial for the maintenance of the near geostrophic balance. Higher-order interpolation yields continuous first derivatives and greatly reduces the noise at the start of the fine grid integrations.

A critical component of the adaptive solution procedure is the ability to set accurately the boundary conditions for the fine grid. During this set of simulations kinks arose in the surface pressure field close to the fine grid boundaries. These kinks are illustrated in Figure 12.

The numerical scheme does not use the pressure gradient at the boundary and, consequently, the kinks have little effect on the solution, because they do not induce

an acceleration which would redistribute the mass and remove the kinks. They do, of course, denote a buildup of mass or a mass deficit in the vicinity of the boundary and indicate a problem with the boundary conditions. These kinks also indicate abnormally large mass divergences close to the boundary. The inclusion of moisture in the model may result in more serious problems. The moisture equations are similar to the surface pressure equation; they are advection equations but they contain source terms. Large mass divergences close to the boundaries may be accompanied by buildups (or erroneous deficits) of moisture. The problem of excess rain close to the boundaries is not uncommon in nested models and is due to an unphysical buildup of moisture next to the the boundaries. In our simple model, kinks in the surface pressure field do not greatly effect the solution, in more complex models the problems which these kinks expose may prove more severe. Hence, next we answer the question of the cause of these kinks and show how they are eliminated.

The inflow and tangential velocities, temperature, and the surface pressure are specified as fine grid boundary conditions by bilinear interpolation from the interior of another grid. As noted in the previous chapter, several investigators have shown that in order to maintain the overall accuracy of the solution the boundary values must be interpolated with an interpolation scheme of the same order accuracy as the numerical scheme and in some cases of one order higher accuracy. Thus we might expect that the problem is an inaccurate specification of the inflow velocity and that a more accurate, higher order interpolation scheme will remove the source of the error and the kinks in the surface pressure field. Tests using bilinear and bicubic interpolation to set the boundary conditions showed very little change in the solution and no decrease in the kinks.

The adaptive scheme attempts to minimize boundary errors in a manner not connected to the structure of the numerical scheme. First, the fine grids are made large enough so that their boundaries are in regions of small solution error, and hence the interpolated values will have small error, even if the interpolation is of low order. Second, if the region of high error cannot be covered by a single grid then multiple overlapping grids are used. An important addition to this is that the fine grid boundary values must be interpolated from the best source, which is often another overlapping fine grid. On the fine grids even low order interpolation will be sufficiently accurate.

Investigation reveals that the kinks in the surface pressure field arise from using a numerical scheme which is inconsistent close to the boundary. One row in from the boundary the diffusion is second order as opposed to the interior fourth order diffusion. This early version of the code also uses a split–explicit scheme for advancing the gravity waves, thereby allowing the use of larger time steps. The split–explicit scheme is also not consistent next to the boundary. Use of the scheme along with the change in diffusion next to the boundaries was found to promote growth of the kinks.

The solution to this problem is to use a scheme which is consistent throughout the entire domain. We accomplish this by making two changes. First we interpolate values for the variables along the boundary and also one row in. This allows the use of the fourth order diffusion. Second, we use a fully explicit scheme. Another important reason to use the fully explicit scheme is to allow interpolation of boundary values from overlapping grids. Presented next are results from the simulation of an unstable baroclinic jet where we use the fully explicit solver. Errors in the specification of the inflow velocity, and hence errors in the divergence and pressure fields, are greatly reduced.

4.2 Baroclinically Unstable Jet

For our second test case we simulate the evolution of an unstable baroclinic jet which has been subjected to an initial perturbation. The disturbance which develops is commonly observed in the atmosphere and easily simulated in a channel. The flow's close analog in the atmosphere and its three dimensional nature allow testing of several adaptive code components untested in the simulation of the two dimensional barotropic flow case.

Previous simulations of this flow were performed in order to gain an understanding of the basic physical processes. Several investigators in the late sixties and early seventies (for example Williams 1967, Mudrick 1974) used models based on the primitive equations and the quasi-geostrophic equations to simulate the developing baroclinic disturbance and the frontal zones associated with it. The cold and warm fronts have received extensive analytic study, most notably in Hoskins and Bretherton (1972). Those interested in the dynamics of this flow can consult these papers or for a more recent and general overview consult Holton (1982).

For this simulation, we solve the primtive equations on a β-plane (Coriolis parameter $= f = f_o + \beta y, \beta = \partial f/\partial y =$ constant). In the following simulations $f_o = 1.479 \times 10^{-5} s^{-1}$, $\beta = 1.748 \times 10^{-11} m^{-1} s^{-1}$ and $0 \leq y \leq 8640 \times 10^3 m$. The grid has five layers at $\sigma = 0.1, 0.3, 0.5, 0.7$ and 0.9. The channel has a west-east length of 5220km. and a north-south width of 8640km. . The north-south velocity v is initially zero and the jet has no variation in the east west direction. The thermodynamic fields are found by requiring that the jet be geostrophically balanced. The balance is derived from equation 3.3 and is given by

$$-\frac{\partial(\pi\phi)}{\partial y} - (RT - \phi)\frac{\partial \pi}{\partial y} - f\pi u = 0.$$

The temperature and geopotential are linked through the hydrostatic equation

$$\frac{\partial \phi}{\partial \ln \sigma} = -RT.$$

As a final constraint we require that the atmosphere be statically stable. Static stability was discussed in Chapter 3. This constraint is satisfied by requiring that the potential temperature θ always increase with height. Plots of the initial conditions can be found in Appendix 1.

The geostrophically balanced jet is perturbed by altering the north-south velocity v. After several simulated days a single dominent wave appears in the channel. The length of the channel is then tripled to $L = 15660$km and the wave repeated twice. By lengthening the channel we force the adaptive code to use multiple overlapping fine grids - as it might in actual atmospheric prediction work. We can also see if the three identical disturbances remain identical, as ideally they should. We note here that a large part of the domain will be covered with fine grids. It is expected that in operational use a much smaller part of the domain will be covered with fine grids.

The initial conditions for the adaptive simulation are shown in Figures 13–16. The jet core, which contains the maximum jet velocities, is found on the $\sigma = 0.3$ layer. The wave is clearly present in the plot of the absolute vorticity ($\psi_a = \psi + f$) on the $\sigma = 0.5$ layer (Figure 13). This primary circulation is the result of the baroclinic instability which arises from the vertical shear present in the jet.

Secondary circulations (vertical and divergent motions) are driven by absolute vorticity advection and temperature advection in the primary circulation and are a result of the hydrostatic and geostrophic nature of the flow. Positive vorticity advection occurs to the west of the trough and negative vorticity advection to the east. In the lower layers, regions of cold and warm temperature advection are found at the trough and crest of the developing wave respectively. The temperature advection can be seen clearly by considering Figure 14, the temperature at the $\sigma = 0.9$ level and Figure 15, the winds at the same level. Horizontal shear and horizontal deformation promote the growth of the cold and warm fronts. The shear and deformation fields intensify in the flow which develops with the development of the surface pressure lows and highs. These surface pressure features are shown in Figure 16. Cyclonic and anticyclonic circulations form at the lower levels around the surface pressure lows and highs. Ageostrophic winds create mass convergence and divergence near the surface at the lows and highs. These circulations are not present in the upper layers of the flow.

Poor representation of the fronts and/or of the jet stream lead to slower development or even decay of the disturbance. The strength and development of these features determines the adequacy of the grid resolution. Figures 17 through 19 show the results after three days of simulation time starting from the fields of Figures 13–

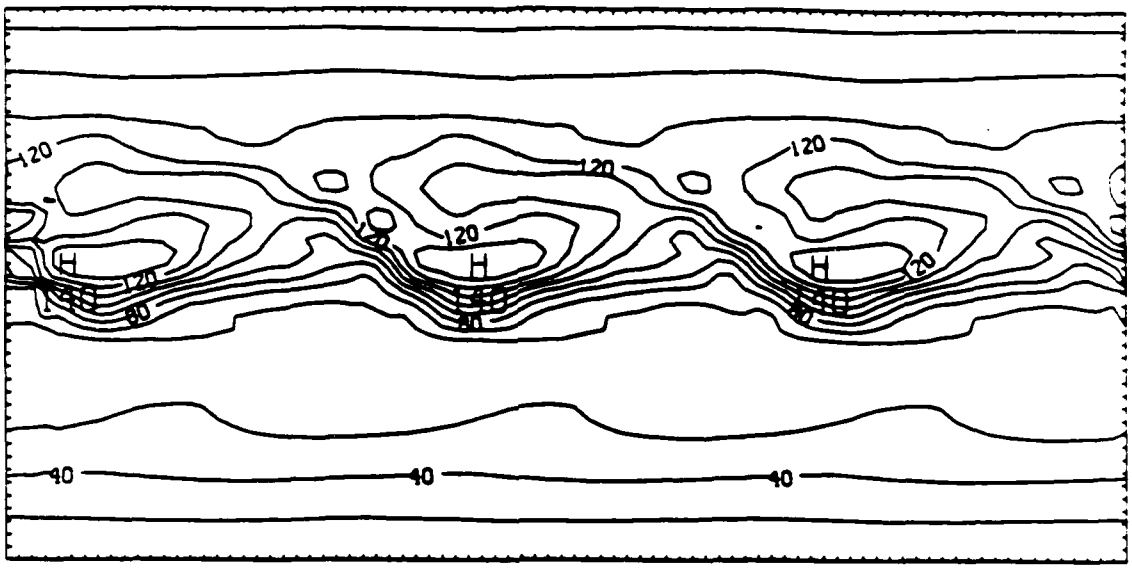


Figure 13 Absolute vorticity ($10^{-6}s^{-1}$) on the $\sigma = 0.5$ surface for the baroclinically unstable jet at $t = 0.$ of the adaptive run.

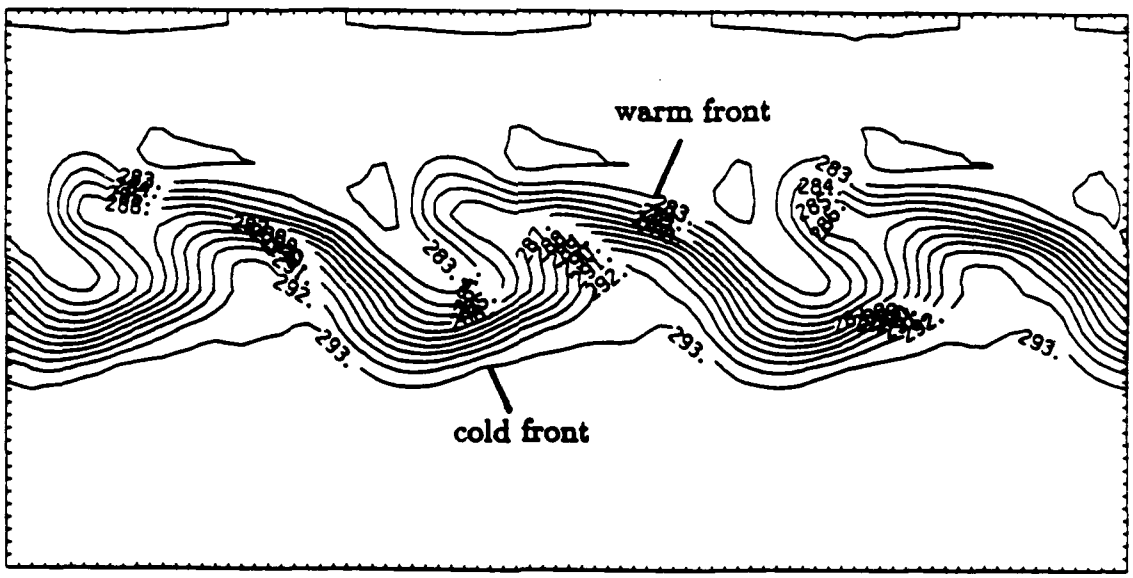


Figure 14 Temperature (K) on the $\sigma = 0.9$ surface at $t = 0$. Locations of the warm and cold fronts are shown.

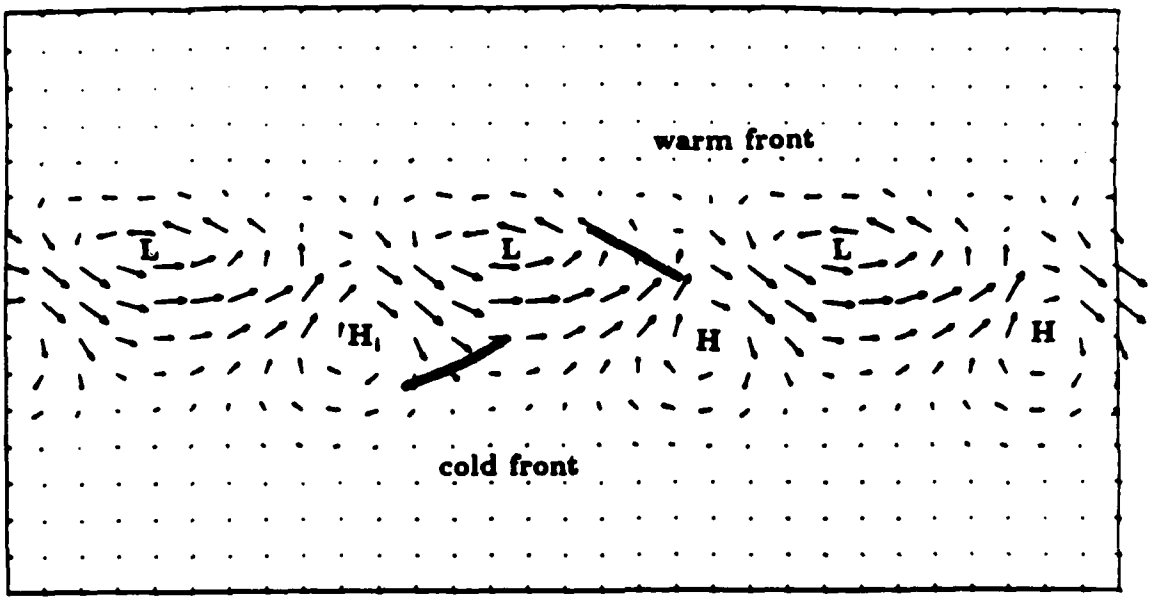


Figure 15 Velocity vectors on the $\sigma = 0.9$ surface at $t = 0$. Note the positions of the warm and cold fronts (shown in Figure 14) and the positions of the surface pressure highs and lows (shown in Figure 16) with respect to the wind fields.

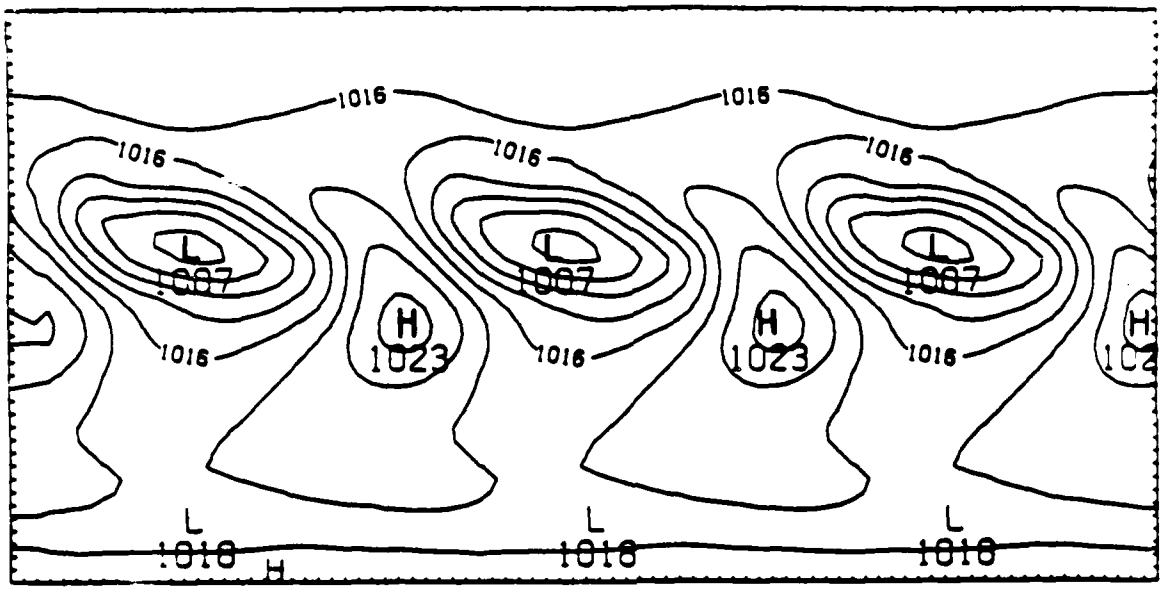


Figure 16 Surface pressure (mb) for the baroclinically unstable jet at $t = 0$.

16 and encompass three different runs; a coarse grid run ($\Delta x = \Delta y = 540$km.), a fine grid run ($\Delta x = \Delta y = 180$km.) and an adaptive run with a coarse grid $\Delta x = \Delta y = 540$km., one level of refinement and a refinement ratio of three (hence $\Delta x_{fine} \approx \Delta y_{fine} \approx 180$km.). Figure 17 shows the surface pressure for the coarse, fine and adaptive grid runs. On the coarse grid the surface pressure highs and lows have lost strength whereas they have not for both the fine and adaptive runs. The warm and cold fronts exhibit similar behavior. The coarse grid fronts are weakening while in the fine and adaptive grid runs they are strengthening. The coarse grid cannot adequately represent the shearing motions and the deformation of the temperature field in the vicinity of the fronts. Again we see that the coarse grid cannot adequately represent the flow while both the fine and adaptive calculations represent the developing baroclinic disturbance well. The same resolution problem is seen in the upper level flow. The maximum absolute vorticity associated with the jet has grown from $1.4 \times 10^{-4}\mathrm{s}^{-1}$ to $1.5 \times 10^{-4}\mathrm{s}^{-1}$ after three days for both the fine and adaptive grid runs but it has diminished to $1.2 \times 10^{-4}\mathrm{s}^{-1}$ in the coarse grid run. The primary wave is deepening in the fine grid run and the adaptive run but it is being washed out in the coarse grid run.

Examination of the surface pressure fields in Figure 17 shows that the fine grid run results and the base (coarse) grid fields for the adaptive run results do not match exactly. Indeed there are some very noticable differences, and the differences are even more pronounced in the absolute vorticity fields. This is simply because the coarse grid cannot possibly represent all the features that are representable on the fine grid. The base (coarse) grid fields for the adaptive run also show the locations of the fine grids which have been placed based on an error estimate of the velocity fields. The adaptive fine grid fields almost perfectly match those of the fine grid run. Even the vorticity fields (Figure 18), which are sensitive to small errors in the velocity, compare extremely well with the fine grid run solution.

In these simulations the largest errors are associated with the jet. The gridfitting routine fits only a single grid over the jet, but this grid is split into two overlapping grids so that they may be accomodated by the limited workspace in the solver. The error is re-estimated every 24 hours and new grids created but the positions of the fine grids change little over several days.

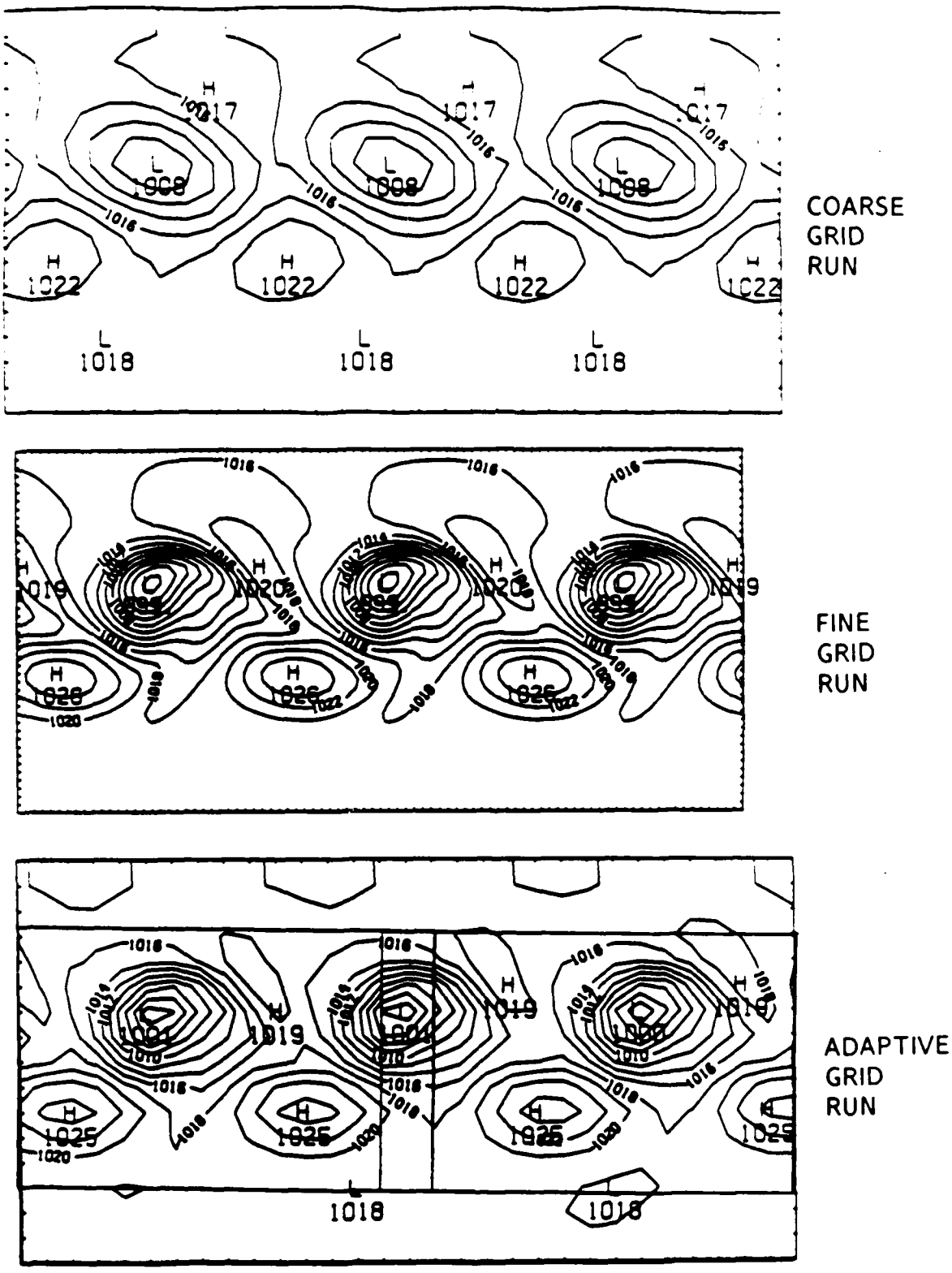


Figure 17 Surface pressure (mb) at $t = 72.$ hours. Locations of the fine grids in the adaptive calculation are shown in the adaptive coarse grid plot.

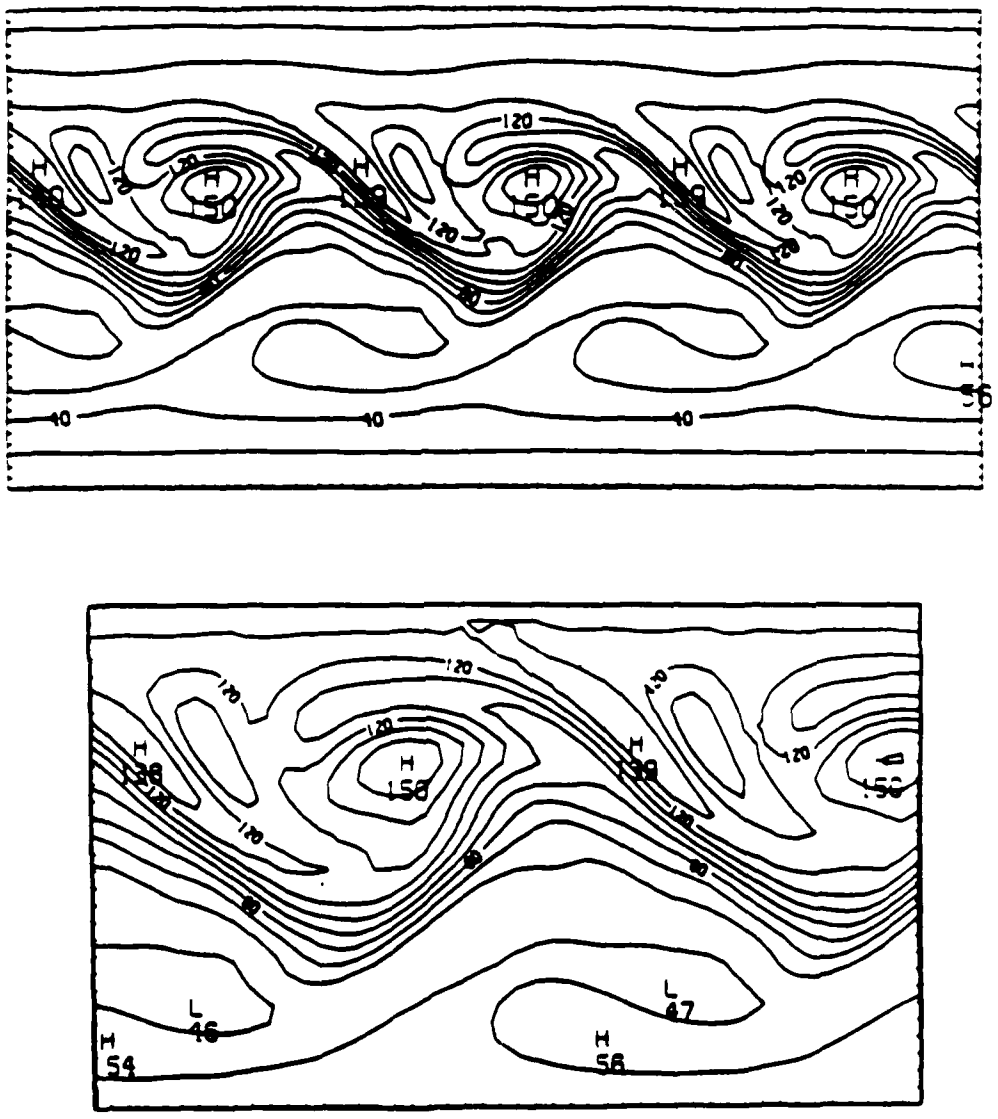


Figure 18 Absolute vertical vorticity on the $\sigma = 0.5$ surface at $t = 72.$ hours for the fine grid run (top) and for the adaptive fine grid 3 (bottom). Adaptive fine grid 3 is the left fine grid in the adaptive run plot in Figure 17. The adaptive fine grid and the fine grid run absolute vorticity fields are almost identical.

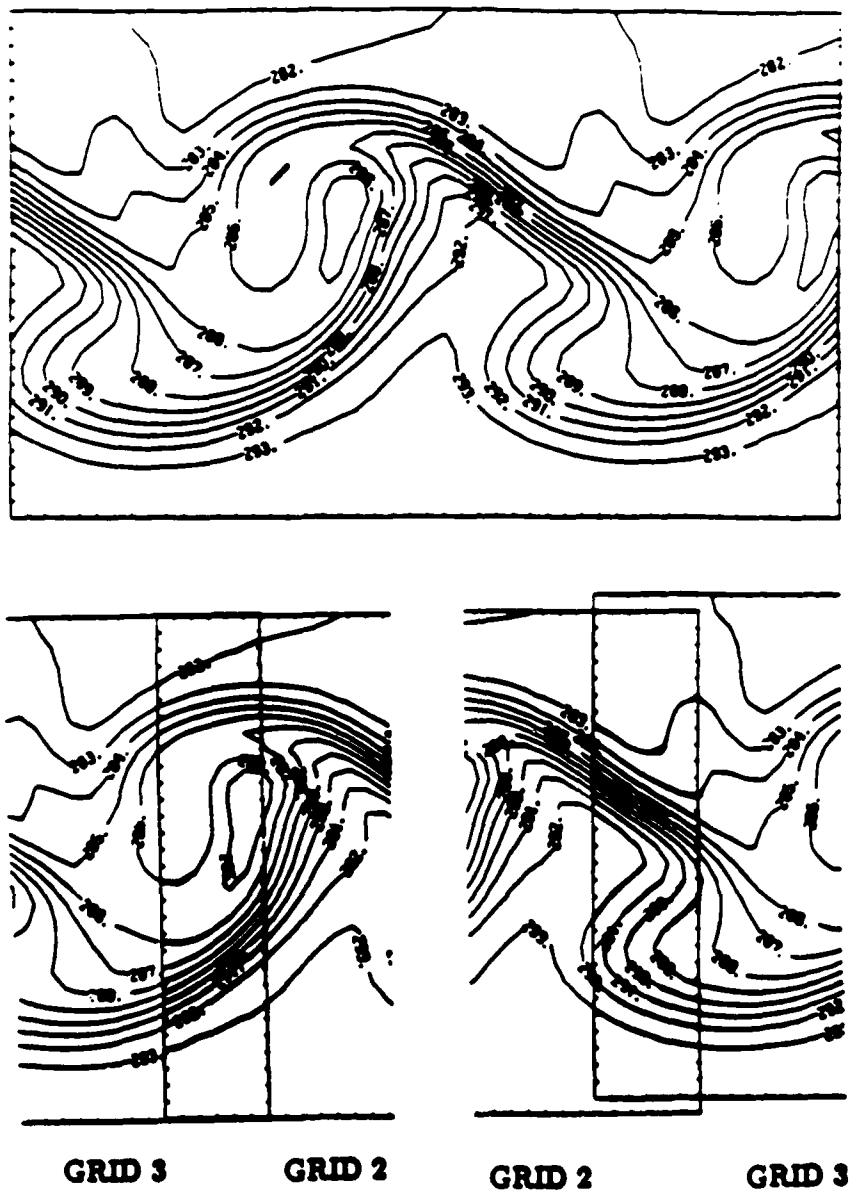


Figure 19 Temperature (K) on the $\sigma = 0.9$ surface at $t = 72.$ hours for the adaptive fine grid 3 (top) and the overlaps between fine grids 2 and 3 (bottom). THe solutions in the overlap regions agree.

Here, as in the previous simulation, the fine grids are not aligned with the base grid. In the overlap region the fine grids are aligned with each other but at the periodic boundary they do not overlap such that their points coincide. In both overlap regions the solutions agree. For this to be the case fine grid boundary conditions must be interpolated from the other fine grid in regions of overlap. Using only boundary conditions interpolated off the coarse grid produces unstable results and is also unsuitably inaccurate.

Surface pressure contours are smooth near the boundary on the fine grids in the adaptive run. In our previous simulation there were kinks in the surface pressure field and large errors in the divergence fields close to the boundary. By using a fully explicit scheme (which allows the setting of boundary condition values from overlapping grids in regions of overlap) and by setting the variables at the first two interior rows as boundary values we have eliminated the kinks and large errors. On the fine grids the disturbances are well represented even in the overlap regions. No noise develops in the overlap region or at the coarse-fine grid boundaries even when these boundaries and overlaps have points which are not coincident. Indeed, the fine grid boundaries and regions of overlap are not readily apparent. Our disturbances remain as three identical disturbances even though different parts of the disturbances are represented in different overlap regions. Figure 19 is a plot of the temperature on the $\sigma = 0.9$ level. Here a cold front and a warm front pass directly through fine grid overlaps and fine grid boundaries. These fronts are identical to those not passing through an overlap region – as they should be.

The numerical scheme used in the solver conserves mass, as do the differential equations. However, the adaptive method used does not explicitly attempt to conserve mass. Variables on the base (coarse) grid are updated where possible with an averaged value from a fine grid and values for boundary conditions are obtained using bilinear interpolation. In Chapter 5 we will show that this formulation very nearly conserves mass, especially when compared with higher order interpolation methods. For a six day fine grid run the maximum mass fluctuation in the channel is approximately .007% of the total mass in the channel. The adaptive and coarse grid runs have fluctuations of .02%. The error in the mass is on the order of a few tenths of a percent at most. The magnitude of this error is comparable to the truncation error of the numerical

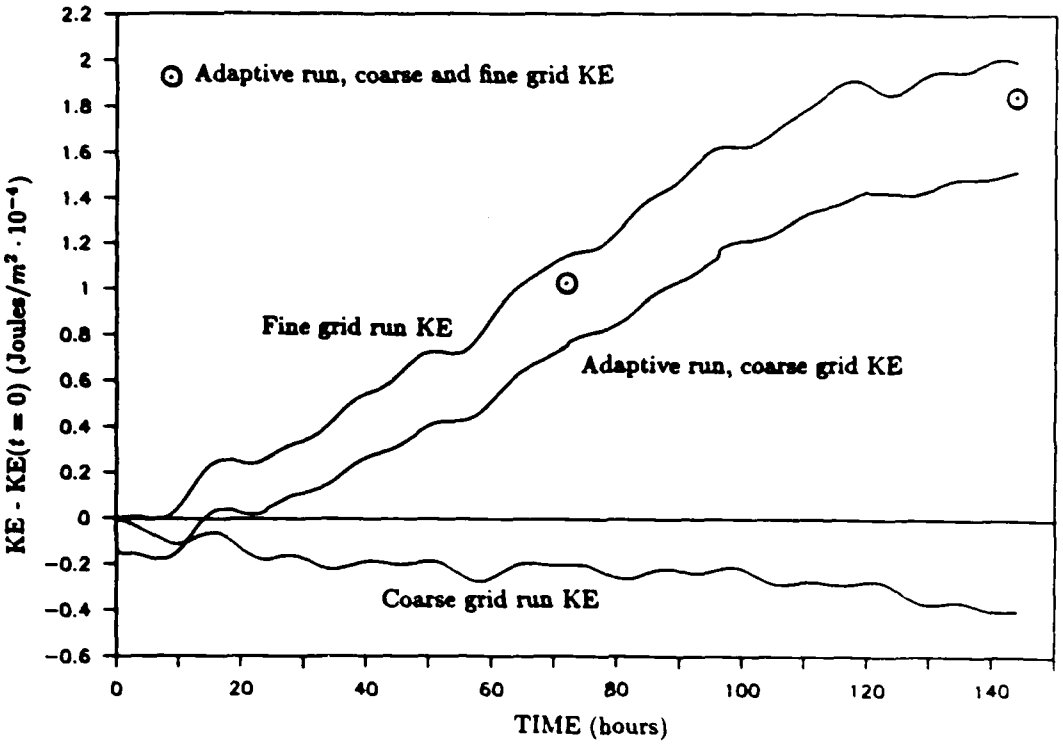


Figure 20 Horizontally averaged kinetic energy.

scheme.

We can also derive conservation equations for kinetic energy (KE) and total potential energy (TPE). The conservation equations show that the total energy (KE + TPE) is conserved in an adiabatic flow. Figure 20 is a plot of the average KE of the atmosphere versus time. The integration is carried out on the base grid during the adaptive run because the changing locations of the fine grids make an "adaptive" integration difficult. The kinetic energy should increase as the disturbances grow and in the fine and adaptive grid runs it does. One should also note that the oscillations are similar for the fine and adaptive runs. We have calculated the KE which includes the contribution from the fine grids in the adaptive grid run at 3 days and at 6 days. These agree well with the fine grid values.

The total energy in the channel (KE+TPE) is dominated by the TPE. In fact, only approximately 0.5% is available for transfer from TPE to KE. Since we solve an adiabatic set of equations with no energy sources or sinks and that our flow is in an enclosed channel, we should find that the total energy is constant over time. This is

not the case, but in all cases the energy in the systems increases, but the increase is less than a tenth of a percent of the average TPE, again on the order of the truncation error in the scheme.

One last run was made using two levels of refinement. Figure 4 illustrates the grid arrangment at 20 hours. Regridding at level 2 was performed every 12 hours and level 3 every 4 hours (refinement ratio $r = 3$). The coarse grid has $\Delta x = \Delta y = 540$km., the first level of refinement has $\Delta x \approx \Delta y \approx 180$km. and the second level of refinement has $\Delta x \approx \Delta y \approx 60$km. Again the refinement is found to be needed around the jet and the maximum errors are at the jet core. A single fine grid run with $\Delta x = 60$km., i.e., of the resolution of the fine grids in the adaptive run, indicates that the increase in resolution from 180km. to 60km. was unnecessary. We performed this integration as a test using multiple levels of refinement. All general conclusions found for the two-level refinement case hold when using three levels.

More results for the calculations described in this section can be found in Appendix 1.

4.3 Error Estimation

Fine grids are placed in the solution domain based on an estimate of the truncation error. The procedure used to estimate the truncation error is based on Richardson extrapolation and it has been described in chapter 2. We wish to demonstrate that the Richardson procedure produces accurate estimates of the truncation error and do this by comparing the Richardson estimate with direct estimates of the truncation error.

The primary advantage of the Richardson based error estimate technique is that the form of the truncation error need not be known. The form of the truncation error associated with the discretized equations 3.1-3.5 is complex and difficult to derive. Also the leading order truncation error terms consist of higher order derivatives which are difficult to compute with more than first order accuracy. The truncation error estimate obtained using equation 2.2 with a qth order method is accurate to $O\big(k(k^{q+1} + h^{q+1})\big)$ which for a second-order scheme such as the one used in the present solver produces a third-order accurate estimate of the truncation error.

The error estimate for the u velocity field in the barotropic cyclone case (Figure 11) show that the regions of high error are around the cyclone. The coarseness of the grid precludes any deeper analysis. To further examine the error estimates we have computed the truncation error for the fine grid run at time $t = 72$ hours using both the Richardson based method and the discretized forms of the leading order terms of the exact truncation error. The error estimates on the fine grid contain detail which cannot be represented on coarser grids.

First we examine the equations and directly estimate the size of the truncation error. Our scheme is second-order in both space and time. Small time steps are used in the explicit scheme due to the presence of the fast gravity waves. Thus, we expect that the dominent truncation error will arise due to the spatial discretization employed and hence we will focus on the error in the spatial differencing. Later comparison of the spatial truncation error with the total truncation error shows that the spatial error does dominate. We can estimate the size of the truncation error by first scaling and then nondimensionalizing equations 3.1-3.5 along with the truncation error. The scaling and nondimensionalization of equation 3.1, the u momentum equation, along with the spatial truncation error, is contained in Appendix 2.

For large-scale atmospheric flows we find that the pressure gradient and Coriolis forces must balance each other and that the advection terms are an order of magnitude smaller than these. This well known result describes the geostrophic nature of the atmosphere, i.e., the approximate balance between the pressure gradient and Coriolis forces. Large-scale flows can often be considered in terms of adjustments to maintain an approximately geostrophic and hydrostatic state.

The finite difference scheme used to discretize the equations is second order accurate: the leading order truncation error term is $O\big(k(k^2+h^2)\big)$. This truncation error is the sum of the truncation errors for the individual terms, all of which are of second order. In the nondimensional equations the pressure gradient and Coriolis terms have coefficients of $O(10)$ while the advection terms have coefficients of $O(1)$. If we look at the order of accuracy of the scheme we might expect that the truncation errors for the Coriolis and pressure gradient discretizations would be an order of magnitude larger than those of the advection terms. This is not the case.

The leading order terms in the nondimensional truncation errors for the Coriolis, pressure gradient and advection terms are:

$$\tau(f\pi v) \approx \frac{5\Delta x^2}{4L^2}(v'_{x'x'} + v'_{y'y'}) \tag{4.1a}$$

$$\tau(\partial(\phi\pi)/\partial x) \approx \frac{5\Delta x^2}{12L^2}(\phi'_{x'x'x'}) \tag{4.1b}$$

$$\tau(\partial(\pi uu)/\partial x) \approx \frac{\Delta x^2}{2L^2}(\frac{2}{3}u'_{x'x'x'} + u'_{x'x'}u'_{x'}) \tag{4.1c}$$

$$\begin{aligned}\tau(\partial(\pi uv)/\partial y) \approx \frac{\Delta x^2}{4L^2}(&\frac{1}{6}u'v'_{y'y'y'} + \frac{1}{2}u'_{y'}v'_{y'y'} + u'_{y'y'}v'_{y'} + \\ &\frac{2}{3}v'u'_{y'y'y'} + u'v'_{x'x'y'} + u'_{y'}v'_{x'x'})\end{aligned} \tag{4.1d}$$

All primed variables are dimensionless and of $O(1)$ and their derivatives are of $O(1)$; Thus, all the truncation error terms are of the same relative size. Direct computations of these terms confirm this. Thus it appears we cannot ignore any of the terms when computing the truncation error for the equations directly.

Figure 21 is a plot of the truncation error in the u velocity field at $\sigma = 0.3$ (jet core) at $t = 72$ hours calculated using the Richardson based technique. Figure 22 is a plot of the truncation error associated with the spatial discretization of the pressure gradient, Coriolis and advection terms computed using the results in Appendix 2. It is

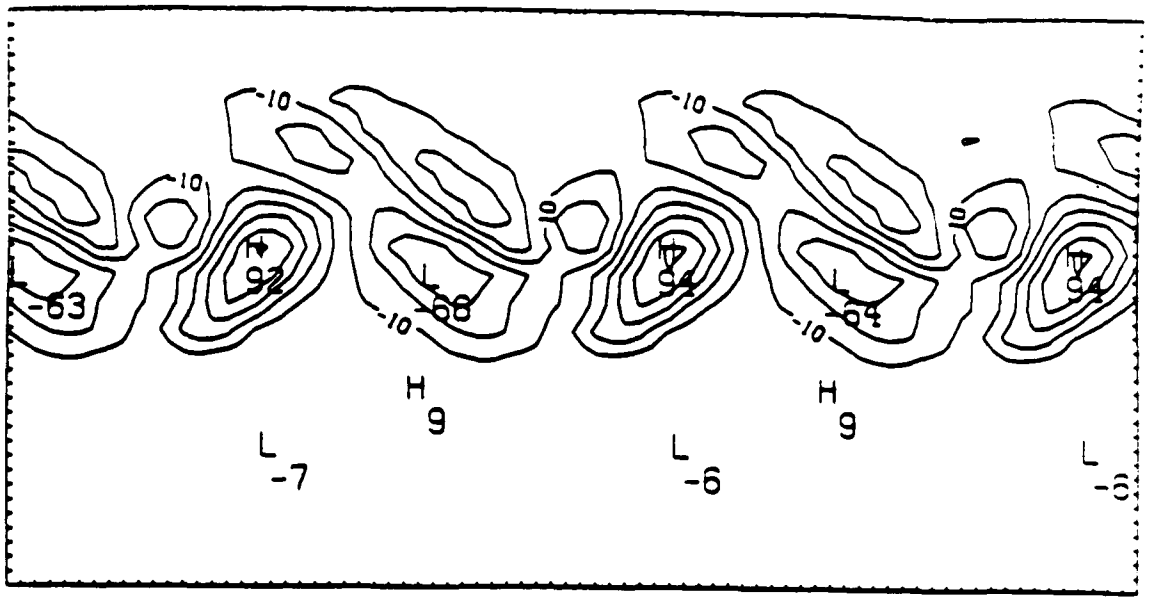


Figure 21 Truncation error estimate using equation 2.2 for the u velocity field at $t = 72.$ hours on the $\sigma = 0.3$ surface. The estimate has been normalized by dividing by $U = 10m/s$ (dimensionless, $\times 10^5$).

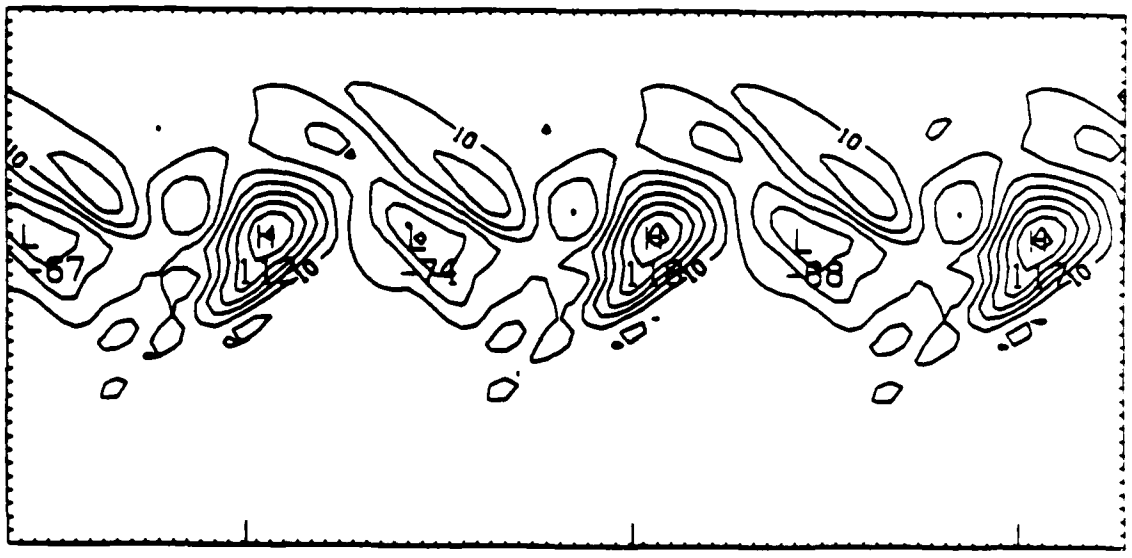


Figure 22 Truncation error estimate computed from the formulas in Appendix 1 (dimensionless $\times 10^5$). Normalized and nondimensionalized by multiplying by $\Delta t/(\pi U)$.

nondimensionalized by multiplying by $\Delta t/(\pi U)$, i.e., in the same way the Richardson estimate is nondimensionalized. The two estimates compare very well. Both the magnitude and distribution of the error are accurately predicted by the Richardson technique. We originally assumed that the spatial truncation error dominated the overall truncation error. This comparison indicates that the assumption is correct.

We have also estimated the error in the surface pressure and temperature fields. The errors in temperature field are large in the cold and warm fronts of the the lower levels. Here the temperature advection is large as is the deformation of the temperature fields. The error estimates for the temperature fields contain significantly more noise than the error estimates for the wind fields. The estimates appear qualitatively correct. It should also be noted that the errors in the temperature are small relative to the scaled temperature; thus regions of high error may have little significance when compared with the overall solution error.

The error estimates for the surface pressure are very noisy. It is very difficult to associate regions of high error with some solution feature. As with the temperature error estimate, the error in the surface pressure is small relative to the scaled surface pressure and the significance of regions of high error may be small, but the small relative size of the error does not explain the large amount of noise in the estimate.

The reason for the noise becomes apparent after the surface pressure tendency equation and its truncation error are scaled. The discretized form of the surface pressure tendancy equation can be found by combining equations 3.6 and 3.7. The continuous equation is

$$\frac{\partial \pi}{\partial t} = -\int_1^0 \nabla_\sigma \cdot (\pi V_h) \partial \sigma$$

and its discretized form is

$$\frac{\pi_{i,j}^{t+\Delta t} - \pi_{i,j}^{t-\Delta t}}{2\Delta t} = -\sum_{k=1}^{K} \frac{1}{2}\big(u_{i+\frac{1}{2},j,k}(\pi_{i,j} + \pi_{i+1,j}) - u_{i-\frac{1}{2},j,k}(\pi_{i,j} + \pi_{i-1,j}) + v_{i,j+\frac{1}{2},k}(\pi_{i,j} + \pi_{i,j+1}) - v_{i,j+\frac{1}{2},k}(\pi_{i,j} + \pi_{i,j+1})\big).$$

The right hand side is evaluated at time t. The dominant error in this discretization will be contained in the discretization of the mass divergence $\nabla_\sigma \cdot \pi V_h$. If we expand the discretization in a Taylor series we find that the truncation error associated with

the divergence term is

$$\tau(\nabla_\sigma \cdot \pi V_h) = \frac{h^2}{6}\left(u\frac{\partial^3 \pi}{\partial x^3} + v\frac{\partial^3 \pi}{\partial y^3} + \pi\frac{\partial^3 u}{\partial x^3} + \pi\frac{\partial^3 v}{\partial y^3}\right) - \frac{h^2}{4}\left(\frac{\partial u}{\partial x}\frac{\partial^2 \pi}{\partial x^2} + \frac{\partial \pi}{\partial x}\frac{\partial^2 u}{\partial x^2} + \frac{\partial v}{\partial y}\frac{\partial^2 \pi}{\partial y^2} + \frac{\partial \pi}{\partial y}\frac{\partial^2 v}{\partial y^2}\right) + O(h^3)$$

where $\Delta x = \Delta y = h$. We can scale the mass divergence and its truncation error with the scaling parameters found in Appendix 3. The dominant terms in the truncation error and their scalings are

$$\frac{h^2}{6}\Big(\underbrace{\pi\frac{\partial^3 u}{\partial x^3}}_{\frac{\pi_o U}{L^3}} + \underbrace{\pi\frac{\partial^3 v}{\partial y^3}}_{\frac{\pi_o U}{L^3}}\Big).$$

We can rewrite the mass divergence as

$$\nabla_\sigma \cdot \pi V_h = \pi \nabla_\sigma \cdot V_h + V_h \cdot \nabla \pi.$$

First, we must recognize that the divergence of the velocity field does not scale as U/L but rather scales as $R_o^2 U/L$ where R_o is the Rossby number. The scaling for the mass divergence is

$$\underbrace{\pi \nabla_\sigma \cdot V_h}_{\frac{R_o^2 \pi_o U}{L}} + \underbrace{V_h \cdot \nabla \pi}_{\frac{\hat{\pi} U}{L}}.$$

For large-scale atmospheric flows $R_o \approx 0.1$, $\hat{\pi} \approx R_o^2 \pi_o$ and the two terms are of the same size.

If we compare the size of the mass divergence with that of its truncation error we find

mass divergence	*truncation error*
$R_o^2 \frac{\pi_o U}{L}$	$\frac{h^2}{6L^2}\frac{\pi_o U}{L}$.

We see that for $R_o^2 \approx (h/L)^2$ the mass divergence will be of the same relative size as its truncation error. This almost always will be the case for global models and often will be the case for limited area (regional) models. In other words *the error in the mass divergence can be as large as the mass divergence itself.* The Richardson procedure will not give accurate estimates of the truncation error for the surface

pressure if this is the case because the truncation error in the calculation of the mass divergence cannot be separated from the mass divergence itself. The Richardson procedure assumes that the truncation error is small, i.e., that the divergence can be accurately calculated.

The observation that the error estimates for the pressure fields tends to be noisy is not new. In adaptive calculations of the 2-D steady state Navier-Stokes equations, Caruso (1985) does not use error estimates of the pressure field for precisely this reason. Berger and Jameson (1985) do not mention the error estimates they use in their adaptive calculations for the steady state Euler equations. These error estimate limitations have yet to be examined in the light of the more complex equation sets being solved adaptively. It is also unclear what the implications are for a numerical scheme which uses a non-zero quantity that is calculated with a scheme having a truncation error as large as the quantity itself.

5. Vertical Refinement with the Primitive Equations

The adaptive results presented in the previous chapter involve refinement in the two horizontal dimensions. In the introduction we have noted that adaptive refinement in two dimensions has been successfully used to solve many types of equations, including hyperbolic, parabolic and elliptic equations. The primitive equations are hyperbolic in nature in the two horizontal directions but are elliptic in the vertical due to the replacement of the vertical momentum equation with the hydrostatic equation. The success of horizontal adaptive grid refinement for the primitive equations required only minor changes in the method developed for simpler sets of hyperbolic equations. The method's success demonstrates the feasibility of using the method for solving a hyperbolic set of conservation laws in two dimensions and should be viewed as having a firm foundation in the work of those before us.

Next we examine the problems encountered when attempting to implement vertical refinement in an adaptive or nested model. For large-scale atmospheric flows, vertical refinement may help to resolve jet streams, upper level fronts, surface boundary layers and other phenomena. In general, though, increasing horizontal rather than vertical resolution is more important in models used for large-scale atmospheric flows. Vertical refinement will be much more important in adaptive and nested models which solve for smaller scale motions, including non-hydrostatic motions. We can envision hydrostatic equations being the basis for models on coarse grids and non-hydrostatic equations as being solved on successively finer grids.

Vertical refinement for the primitive equations presents a variety of new problems that have not been directly addressed by other investigators. Some of the problems arise due to the elliptic nature of the equations in the vertical, while others are problems that are also present when constructing horizontal refinement methods but which may be more severe in the case of vertical refinement. We have not found compatible solutions for these problems and we do not believe that vertical refinement with the primitive equations, using presently available techniques, is viable. We present these results as an argument for this claim and to aid those who may be attempting to construct a vertically refining adaptive or nested atmospheric model based upon the hydrostatic primitive equations.

The practical problems we've encountered are centered around the necessity to vertically interpolate dependent variables. Three problem areas can be defined; 1) the prescription of conservative boundary conditions for fine grids and conservative updating techniques for the transfer of information from the fine to the coarse grids, 2) the preservation of basic flow balances in the initialization of fine grid fields from coarse grid fields and 3) the conflicting needs of preserving the relative static stability of the atmosphere against the need to carefully interpolate the horizontal pressure gradient. In the next three sections we discuss the three problem areas and present techniques we've tried. We also show how these problems are either solved or do not arise in the construction of a horizontally refined model.

As the character of the equations differs in the horizontal and vertical dimensions so does, of course, the resulting physical nature of the flow. Many of the problems have a physical interpretation that often provides the clearest insight into their nature. We present physical interpretations wherever possible. In the last section of this chapter we present some theoretical results concerning the primitive equations which indicate more fundamental problems may exist concerning the use of the primitive equations in a vertically refining model.

5.1 Interface Conditions

The primitive equations prescribe the conservation of mass, momentum and energy. From this set other equations can be derived which describe the conservation of kinetic energy and total potential energy. Equations of this type exist in descriptions of most fluid flows. The accuracy of numerical solutions can be gauged in part by observing how well mass, kinetic energy and potential energy have been conserved. Many argue that conservative schemes (those which conserve various quantities exactly) produce overall more accurate solutions than non-conservative schemes; however, accurate computation of the solution of a conservative system implies that the conserved quantities will be accurately conserved, but accurate or absolute conservation does not imply that the solution is accurate. Thus, those who are concerned with the numerical solution of fluid flow equations have examined the effect of boundary conditions upon the overall conservation properties of a numerical scheme.

If a numerical scheme exactly conserves some quantity which is shown to be conserved by the continuous equations then in order to maintain that exact conservation the boundary conditions must also exactly conserve the flux of this quantity across an interface. Our adaptive method increases the number of interfaces and the issue of conservation arises in both the setting of boundary conditions for fine grids and in the updating of the coarse grid solutions with the fine grid solution.

Consider the function G which is defined as the integral of some quantity A over the domain Ω. In our calculations A may be the kinetic energy per unit volume, the total energy per unit volume or the mass per unit volume and Ω is some volume of interest. Figure 23 shows a domain subdivided into regions Ω_1 and Ω_2. We can define $G(A)$ in the region $\Omega_1 + \Omega_2$ as

$$G(A) = \int_{\Omega_1} A d\Omega + \int_{\Omega_2} A d\Omega. \tag{5.1}$$

For the quantity A we can write a conservation law of the form

$$\int_{\Omega} \frac{\partial A}{\partial t} d\Omega = \int_{\Omega} S(A) d\Omega + \int_{\Gamma} F(A) \cdot \vec{n}\, d\Gamma \tag{5.2}$$

which states that the time rate of change of the variable A integrated over the region Ω is just the integral of the sources and sinks S of A over the region and the integral

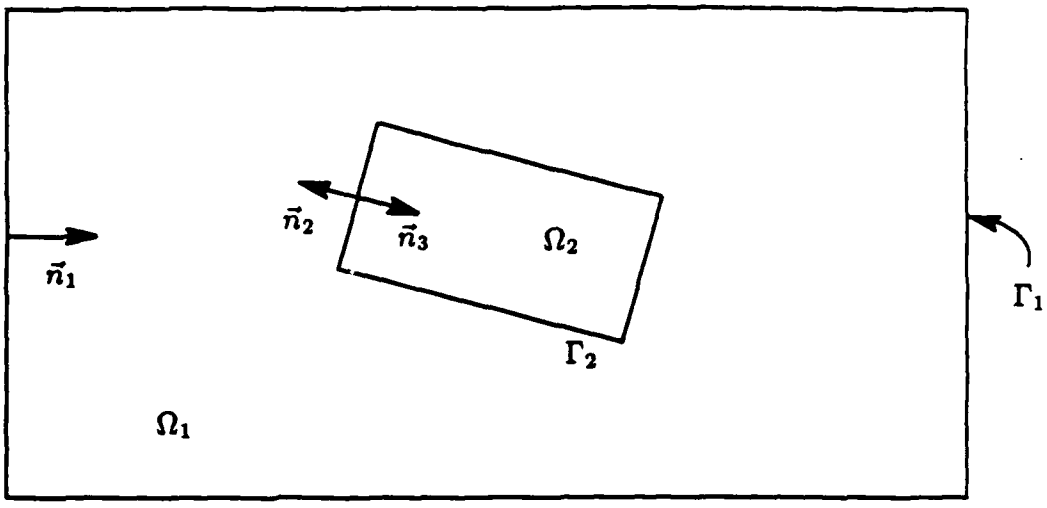


Figure 23 The domain Ω. $\Omega = \Omega_1 + \Omega_2$ and $\Omega_1 \notin \Omega_2$.

of the fluxes F of A through the boundary Γ. Taking the derivative with respect to time of the quantity $G(A)$,

$$\frac{d(G(A))}{dt} = \int_{\Omega_1} \frac{\partial A}{\partial t} d\Omega + \int_{\Omega_2} \frac{\partial A}{\partial t} d\Omega,$$

and substituting for $\partial A/\partial t$ using 5.2 results in

$$\frac{d(G(A))}{dt} = \underbrace{\int_{\Omega_1} S(A) d\Omega}_{term1} + \underbrace{\int_{\Gamma_1} F(A) \cdot \vec{n}_1 \, d\Gamma}_{term2} + \underbrace{\int_{\Gamma_2} F(A) \cdot \vec{n}_2 \, d\Gamma}_{term3}$$
$$+ \underbrace{\int_{\Omega_2} S(A) d\Omega}_{term4} + \underbrace{\int_{\Gamma_2} F(A) \cdot \vec{n}_3 \, d\Gamma}_{term5}. \tag{5.3}$$

Terms 1 and 4 are sources of A in regions 1 and 2. Terms 2, 3 and 5 are fluxes of A through the boundaries. Terms 3 and 5 should be equal but of opposite sign because the flux is being computed at the same boundary and only the sign of the normal vector changes.

We can use this model to examine the mass conservation properties of the adaptive scheme. If we imagine that the region $\Omega_1 + \Omega_2$ of Figure 23 defines a coarse grid and the region Ω_2 defines a fine grid then the interpolation and averaging schemes used in setting boundary conditions and in updating may affect the terms on the right

hand side of 5.3 and the calculation of $G(A)$. We want our method to maintain the conservation principle defined in 5.3.

The numerical scheme in the primitive equations model exactly conserves mass, meaning that mass is conserved up to machine accuracy. Terms 1 and 4 are both equal to zero. But when we compute 5.1, we actually compute

$$G(A) = \int_{\Omega_1} A d\Omega + \int_{\Omega_2} \tilde{A} d\Omega$$

where $\tilde{A}$ represents the averaged value from the fine grid solution that replaces the coarse grid solution in the updating process. For mass conservation A denotes the surface pressure π and the domain Ω is the horizontal area. We use an area average of the surface pressure over the scale of the coarse grid when updating the surface pressure on the coarse grid; hence, mass is still conserved exactly and the source terms 1 and 4 are equal to zero.

Terms 3 and 5, the mass flux through the boundary Γ_2, will cancel if the mass flux is conserved in the interpolation used for setting the fine grid boundary conditions and in the averaging scheme used in the updating procedure. The spatial discretization of the equation representing mass conservation (3.5) is

$$\frac{\partial \pi}{\partial t} = \sum_{k=1}^{K} \nabla \cdot (\pi V_k) \Delta \sigma_k \tag{5.4}$$

where

$$\begin{aligned}\nabla \cdot (\pi V_k) =& \frac{1}{2\Delta x}\left(u_{i,j,k}(\pi_{i,j} + \pi_{i+1,j}) - u_{i-1,j,k}(\pi_{i-1,j} + \pi_{i,j})\right) + \\ & \frac{1}{2\Delta y}\left(v_{i,j,k}(\pi_{i,j} + \pi_{i,j+1}) - v_{i,j-1,k}(\pi_{i,j-1} + \pi_{i,j})\right)\end{aligned}$$

and the notation for the discretization is as in Figure 24. The mass flux across the surface shown in Figure 24 is computed as

$$F = u_{i-1,j,k}(\pi_{i-1,j} + \pi_{i,j}) \cdot \frac{\Delta \sigma_k \Delta y}{2}.$$

First, consider the case where the interface lies between a coarse grid and a horizontally refined fine grid. It is sufficient to consider the mass flux across a line

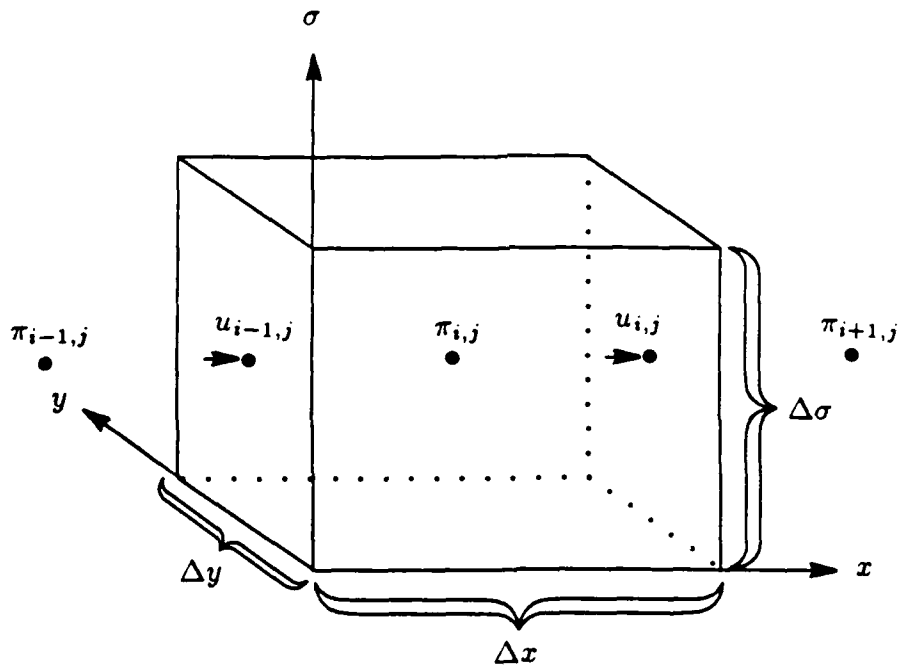


Figure 24 Control volume centered about point $\pi_{i,j}$.

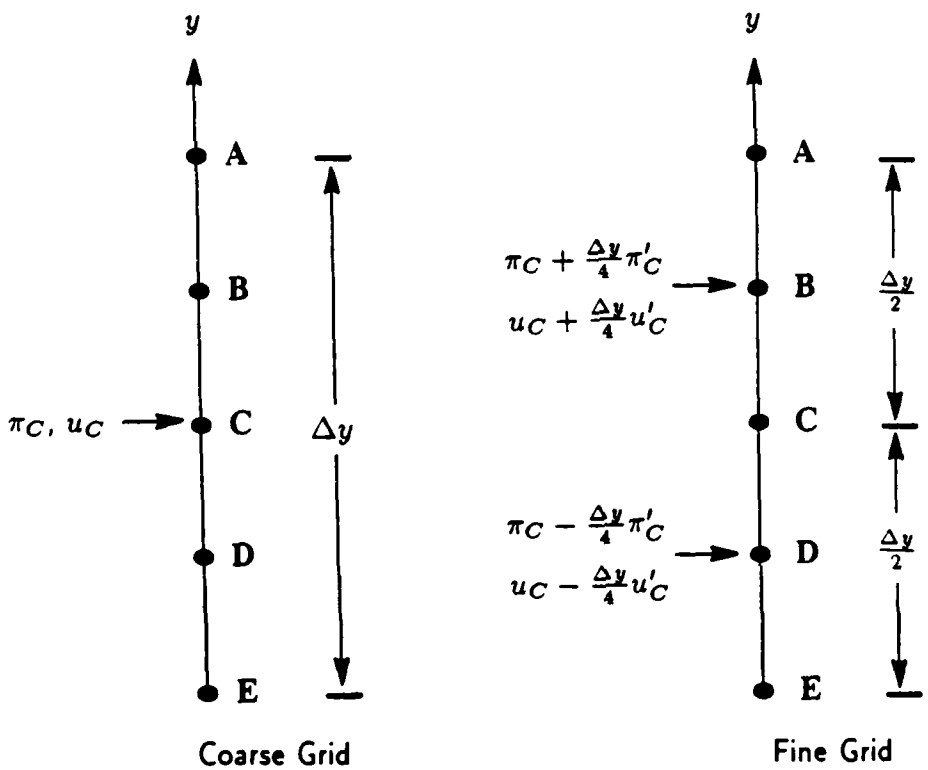


Figure 25

AE as shown in Figure 25. If we expand u and π in a Taylor Series in y about the point C we can compute the mass flux on the coarse grid, F_c, across the line AE as

$$F_c(AE) = u_C \pi_C \Delta y \Delta \sigma_k.$$

In the adaptive grid method, fine grid values for u and π are interpolated from the coarse grid using a first order interpolation scheme. The mass flux across AE as computed on the fine grid is

$$\begin{aligned} F_f(AE) =& F_f(AC) + F_f(CE) \\ =& (u_C + \frac{\Delta y}{4} u'_C)(\pi_C + \frac{\Delta y}{4} \pi'_C) \frac{\Delta y \Delta \sigma_k}{2} \\ &+ (u_C - \frac{\Delta y}{4} u'_C)(\pi_C - \frac{\Delta y}{4} \pi'_C) \frac{\Delta y \Delta \sigma_k}{2} \\ =& (u_C \pi_C + \frac{1}{16} \Delta y^2 u'_C \pi'_C) \Delta y \Delta \sigma_k \end{aligned}$$

where the prime denotes a derivative with respect to y. The error in the mass flux across AE is the difference between the mass flux computed on the coarse grid and that computed on the fine grid. This error is

$$\begin{aligned} F_{error} &= F_c(AE) - F_f(AE) \\ &= -(\frac{1}{16} \Delta y^2 u'_C \pi'_C) \Delta y \Delta \sigma_k. \end{aligned}$$

In our method the variables are interpolated separately; hence, boundary values for the fine grid which are interpolated off the coarse grid do not conserve mass and have this error term. We can immediately note that for a constant surface pressure or a constant normal velocity the interpolated mass flux is exact. For non-constant u and π the error term is non-zero. We can estimate the size of the error compared to the overall mass flux by appropriately scaling u, π and y. The correct scalings are

$$\begin{aligned} u =& U u^* \\ \pi =& P_o(1 + S_1 \pi^*) \\ y =& L y^* \end{aligned}$$

where $U = 10m/s$, $P_o = 1000mb.$, S_1 is a nondimensional scaling parameter with $10^{-2} \leq S_1 \leq 10^{-1}$ for large-scale atmospheric flows, $L = 1000km.$ and the variables with astericks are nondimensional and of order one. Substituting these relations into

the error term and normalizing by the mass flux $F_c(AE)$ gives us a leading order error term

$$F^*_{error} = -(\frac{\Delta y^2}{L^2})\frac{S_1}{16} + O(\Delta y^2 S_1^2).$$

This is usually a small and almost negligible error because derivatives of the pressure do not scale with the average surface pressure P_o but rather with $S_1 P_o$.

If we were to use a higher order interpolation, for example a bi-quadratic scheme, then the dimensional error would be

$$F_{error} = -\frac{\Delta y^2}{16}(u'\pi' + \frac{1}{2}u\pi'' + \frac{1}{2}u''\pi + \frac{\Delta y^2}{64}u''\pi'')\Delta y \Delta\sigma_k$$

and the leading order term in the nondimensional normalized error would be

$$F^*_{error} = -\frac{1}{32}(\frac{\Delta y^2}{L^2}) + O(\frac{\Delta y^2}{L^2}S_1).$$

The leading order term in the error for a quadratic interpolation scheme is larger by a factor of S_1^{-1} than the error term for the linear interpolation and the error exists even if the surface pressure is a constant.

The integration results presented in Chapter 4 indicate that the errors are small when using bi-linear interpolation. Mass is very nearly conserved in the horizontally refining adaptive model. Keeping the fine-coarse grid boundaries in regions where the overall error is small also contributes to minimizing the interpolation error. It should be noted that we could exactly conserve mass by interpolating the mass flux $\pi \cdot u$ as opposed to interpolating π and u separately. We have chosen to use the original method for it introduces only small errors, and interpolating the variables separately helps maintain critical balances. We discuss the latter in Section 5.2 .

Vertical interpolation is necessary when a fine grid has more layers in the vertical than a coarse grid, i.e., when implementing vertical refinement. The computation of the mass flux through a boundary differs between a fine and coarse grid because there are a different number of layers to sum over (see equation (5.4)). However, only the velocities need be interpolated in the vertical; the surface pressure is defined only at the surface. Using bilinear interpolation in the vertical exactly conserves the mass flux. Higher order interpolation schemes result in errors similar to the errors found for higher order schemes in the horizontal. Simple tests indicate that the mass fluxes

may differ by several percent between the coarse and fine grids when using higher order schemes.

Mass conservation is not only an issue when setting fine grid boundary conditions but also when updating a coarse grid solution from a fine grid solution. Terms 3 and 5 of equation 5.3 will cancel only if both the interpolation used in setting the boundary conditions and the averaging scheme used in the updating procedure conserve mass flux.

In the horizontally refining model, the fine grid solution is averaged over the scale of the coarse grid. Thus, for a refinement ratio of n, n^2 neighboring points on the fine grid would be averaged to produce the new coarse grid value in the updating procedure. It can be shown that this procedure will approximately conserve mass flux in much the same manner as the bilinear interpolation serves to conserve mass flux in the assignment of fine grid boundary values. Intuitively, we can think of the mass flux through a coarse grid cell face as being divided over several cells faces of the fine grid during the interpolation used to set fine grid boundary values. In the updating procedure, the mass flux through several fine grid cell faces is averaged and used as the flux through a single coarse grid cell face. In the vertical a similar averaging procedure must be used in order to conserve mass in a vertically refining model.

It is easy to construct such an averaging scheme. A coarse grid cell mass flux could be the sum of the fine grid cell fluxes which lie in the coarse grid cell. An important note is necessary here. The interpolations are not normally reversible except for the simplest of profiles or for a non-staggered grid. For example, if we interpolate, in a mass conserving manner, a fine grid field from an existing coarse grid and then used a mass conserving averaging scheme to compute new coarse grid values from the fine grid values, we find that the new values do not necessarily equal the old values, though the mass flux would be conserved.

Mass conservation is crucial for accurate integrations of the primitive equations. Figure 26 is a plot of the average surface pressure in the test channel during the simulation of a developing baroclinic disturbance. The results are obtained using four vertically refining models with different types of vertical interpolations. All the models use the same initial conditions. The models are differentiated as follows:

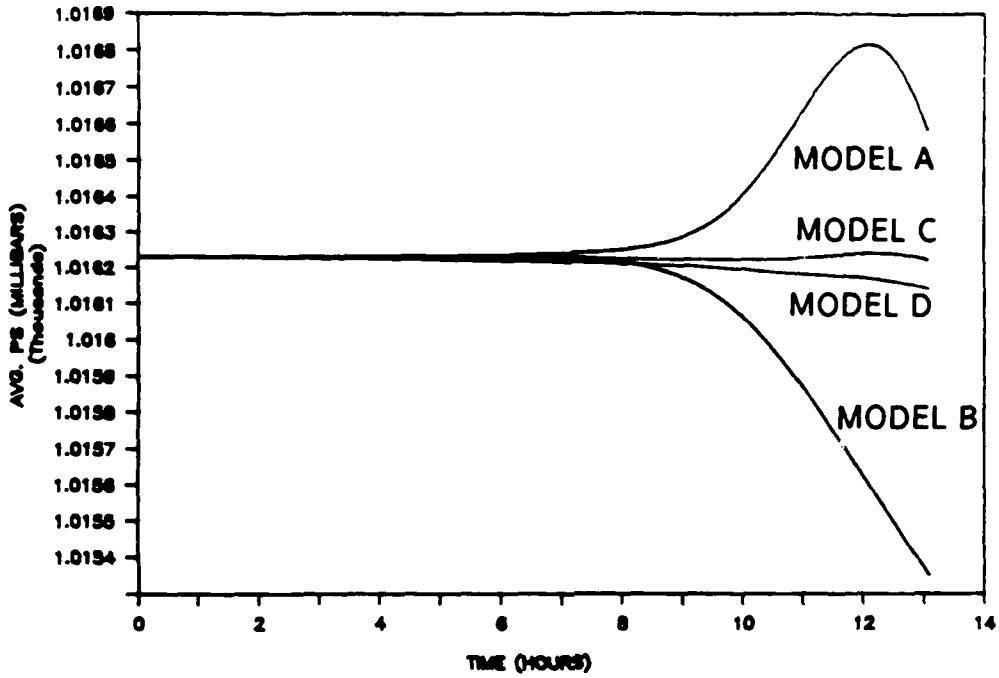


Figure 26 Average surface pressure (mass) for model runs testing fine grid boundary conditions and coarse grids updating procedures in a vertically refining model.

Model	Fine Grid Boundary Condition Scheme	Vertical Averaging for Updating
Model A	quadratic	non-conservative
Model B	linear	non-conservative
Model C	linear	conservative
Model D	quadratic	conservative

The conservative updating scheme is the one previously described in this section and the non-conservatve updating scheme incorporates no averaging, i.e., it replaces the coarse grid value with the fine grid value. Note that in the previous analysis the linearly interpolated boundary conditions were found to be much more closely conservative than quadratic interpolation. While none of the vertically refining model runs was successful (noisy, inaccurate solutions were obtained), the poorest solutions occurred when mass was most poorly conserved. Mass was most poorly conserved when using a

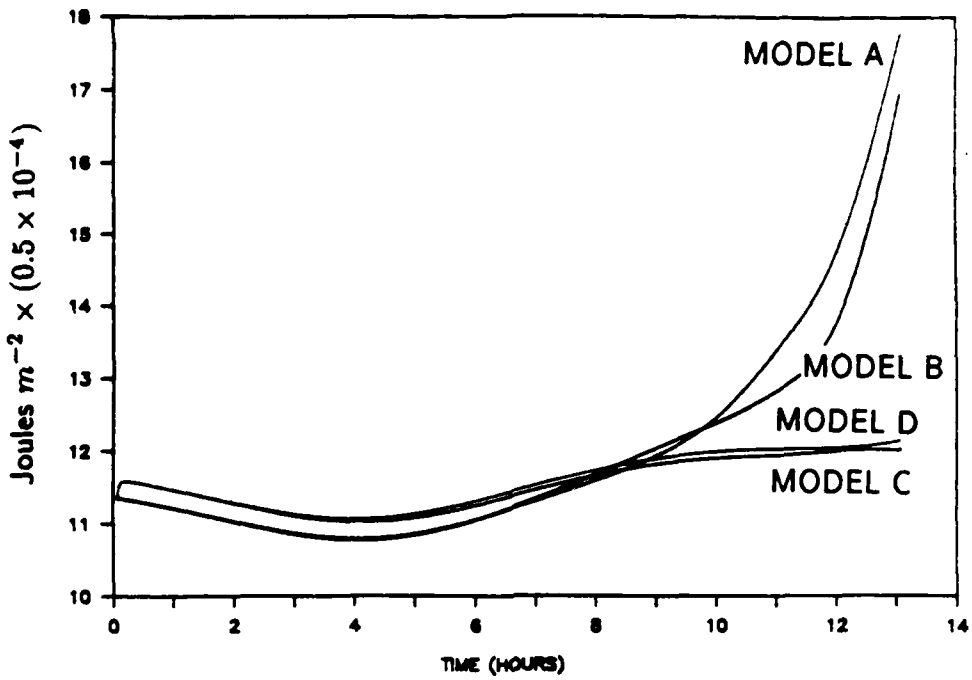


Figure 27 Average kinetic energy for the same runs whose results are shown in Figure 26.

prevent integrations from becoming unstable.

Mass is only one quantity that is conserved by the differential equations. The primitive equations also conserve total energy (kinetic plus total potential). The conservation equations for kinetic energy (KE) and total potential energy (TPE) contain only one source term denoting a transfer of energy between KE and TPE. The equations which describe the conservation of these quantities are of the form (5.1), and boundary fluxes of these quantities can be examined. First order interpolation, which only approximately conserves mass, will not conserve KE. The averaging process used in the updating procedure also will not conserve KE.

Figure 27 depicts the average KE in the channel for the runs used to test mass conservation in a vertically refining model. Kinetic energy is not globally conserved because there is conversion of potential to kinetic energy as the baroclinic disturbance develops. Using the single coarse grid results and the non-vertically refining adaptive results as a guide, we see that mass conservation is necessary for realistic KE time evolution. We cannot predict how the total KE of the flow will evolve when mass is not conserved. We also see jumps in the KE plots at the beginning of integrations.

results as a guide, we see that mass conservation is necessary for realistic KE time evolution. We cannot predict how the total KE of the flow will evolve when mass is not conserved. We also see jumps in the KE plots at the beginning of integrations. These are the result of the smoothing of the coarse grid velocity fields due to the averaging which occurs during the updating procedure. These jumps are also present in the results of Chapter 4. They appear to have very little effect on the solution.

We have observed that fluctuations in the TPE satisfy the same trends as fluctuations in the total mass when mass is not conserved. The calculation for determining the TPE in our channel is

$$TPE = \frac{c_p}{g} \int_X \int_Y \int_{\sigma=1}^{\sigma=0} (T\pi)\ \partial\sigma\partial y\partial x. \tag{5.7}$$

The TPE of a parcel is weighted by it's mass, hence the importance of mass conservation is of no surprise. Note, though, that the KE trend is not similar to the trend in the total mass when the total mass is not conserved.

Mass conservation is crucial. Boundary conditions which conserve mass are necessary if the overall method is to be conservative and integrations stable. Boundary conditions and updating schemes which conserve KE and TPE do not appear to be important, at least for the horizontally refining model. For a vertically refining model other problems, discussed in the next two sections, prove to be more troublesome.

5.2 Initialization

The fields for newly created fine grids are interpolated from coarser grids or from previously existing fine grids. The fields from which we are interpolating already exhibit the balances that characterize large-scale atmospheric flows and need not be filtered or adjusted for numerical integration. These balances are the result of previous integrations or initialization programs which affect this filtering and adjusting. Our goal is to interpolate these fields without altering the balances that exist.

Three observations can be made about large-scale atmospheric flows. The atmosphere is very nearly hydrostatic. It is nearly geostrophic. The divergence of the horizontal velocity field is small. Accordingly, the primitive equations assume a hydrostatic atmosphere and the numerical scheme enforces this. The practical result is that either temperature or geopotential can be interpolated, but not both. Some difficulties associated with this will be discussed in the next section. Here we discuss the preservation of the geostrophic balance and the divergence fields in the interpolations used to initialize new fine grid fields.

The divergence field is used to calculate the surface pressure tendency and the vertical velocity $\dot{\sigma}$. The pressure tendency is computed by integrating (3.5) from $\sigma = 1$ to $\sigma = 0$ and the vertical velocity can then be found by integrating (3.5) from the bottom boundary to the level where $\dot{\sigma}$ is required.

The importance of the geostrophic balance is shown in Appendix 1. The pressure gradient and the Coriolis terms are large compared to the horizontal accelerations but they very nearly cancel each other. Small departures from this balance contribute to driving the large-scale flows.

We are caught in a classic numerical analysis problem. Our computations hinge on the small differences between large numbers. Velocities are large, their gradients are smaller and their divergence, the sum of the gradients, is much smaller than either. Calculations of the pressure gradient term in the geostrophic balance exhibit the same problem.

When refining the grid horizontally, we have found that bilinear interpolation (first order) does not preserve these balances. Gravity waves (noise) emanate from regions of imbalance and render the solutions useless. Higher order interpolations preserve these balances and yield accurate adaptive solutions.

These observations can be easily explained. First order interpolations do not yield continuous first derivatives, but rather yield piecewise constant first derivatives. The balances involve first derivatives and accurate interpolations which yield continuous first derivatives are essential. Higher order interpolations produce continuous first derivatives and at present we use cubic interpolation in the horizontal for initializing fine grid fields.

The geostrophic balance and the calculation of the divergence involve horizontal derivatives and not vertical derivatives. Still, care must be taken when choosing vertical interpolation schemes because these quantities vary vertically. The largest problem arises when attempting to preserve the geostrophic balance. To illustrate this we define a vertical interpolation scheme with weights $W_{p,q}$ and define the vertical interpolation of a variable u on grid 1 (u^1) to grid 2 at level p (u_p^2) as

$$u_p^2 = \sum_{q=1}^{Q} W_{p,q} u_q^1 \tag{5.5}$$

where Q is the number of layers in grid 1. $W_{p,q}$ can be constructed to yield an interpolation scheme accurate to order $Q-1$.

It is sufficient to consider the non-flux form of the geostrophic balance. For the x-momentum equation this relation is

$$-\frac{\partial \phi}{\partial x} - RT\frac{\partial \pi}{\partial x} + fv$$

and is finite differenced on the staggered grid of Figure 6, for level p, as

$$\begin{aligned} &-\frac{1}{\Delta x}(\phi_{i+1,j,p} - \phi_{i,j,p}) - \frac{R}{2\Delta x}(T_{i+1,j,p} + T_{i,j,p})(\pi_{i+1,j} - \pi_{i,j}) \\ &+ \frac{f}{4}(v_{i,j,p} + v_{i+1,j,p} + v_{i,j-1,p} + v_{i+1,j-1,p}) \end{aligned} \tag{5.6}$$

If we use the vertical interpolation scheme (5.5) for the velocities and the geopotential then the balance will not be correctly interpolated. Substituting the interpolation defined by (5.5) into (5.6) yields the resulting balance on a new fine grid for level p.

$$\begin{aligned} &\sum_{q=1}^{Q} W_{p,q}\left(-\frac{1}{\Delta x}(\phi_{i+1,j,q}^1 - \phi_{i,j,q}^1) + \frac{f}{4}(v_{i,j,q}^1 + v_{i+1,j,q}^1 + v_{i,j-1,q}^1 + v_{i+1,j-1,q}^1)\right) \\ &-\sum_{q=1}^{Q} H_{p,q}\left(\frac{R}{2\Delta x}(T_{i+1,j,q} + T_{i,j,q})(\pi_{i+1,j} - \pi_{i,j})\right) \end{aligned} \tag{5.7}$$

Using a method such as this results in imbalances between the pressure gradient and Coriolis force. The ensuing integrations are noisy and the solutions useless. The balance is not directly interpolated because we cannot interpolate the temperature T and the geopotential ϕ independently. T and ϕ are linked through the hydrostatic equation (3.3). In the above example we interpolated the geopotential and used the hydrostatic equation to compute the temperature. This new interpolation is denoted by H in (5.7). We can define H explicitly. When we integrate the hydrostatic equation (3.3) to determine ϕ from T then we can write this integration as

$$\phi = A_P T \tag{5.8}$$

where ϕ and T are column vectors with P entries corresponding to the geopotential and temperatures in a vertical column and A is a $P \times P$ matrix defining the integration of (3.3). We can define the interpolation of T from a coarse grid at Q levels (T_1) to a vertically refined grid with P levels (T_2) as

$$T_2 = \left(A_P^{-1} W A_Q\right) T_1.$$

Here W is a $P \times Q$ matrix. H is simply the matrix defined by the right hand side.

$$H = A_P^{-1} W A_Q \tag{5.9}$$

Note that H is a $P \times Q$ matrix. We will say more about this matrix in Section 5.3.

What is needed is a scheme in which the interpolation of the individual variables results in an interpolation of the balance. The current form of the equations precludes such a scheme. A simple approximation that would allow use of this set in a vertically refining model would be to replace T in the horizontal pressure gradient term with $\tilde{T}(\sigma)$. We can write the pressure and density as

$$p = P_o[p_o(\sigma) + S_1 p']$$
$$\rho = R_o[\rho_o(\sigma) + S_1 \rho']$$

and as before $10^{-2} \leq S_1 \leq 10^{-1}$. Using the equation of state we can rewrite the term RT as

$$\begin{aligned} RT =& \frac{P_o}{R_o} \frac{(p_o(\sigma) + S_1 p')}{(\rho_o(\sigma) + S_1 \rho_i)} \\ =& \frac{P_o}{R_o} \frac{p_o(\sigma)}{\rho_o(\sigma)} (1 + S_1 \frac{p'}{p_o(\sigma)}) \cdot (1 - S_1 \frac{\rho'}{\rho_o(\sigma)} + O(S_1^2)). \end{aligned}$$

If we define $R\tilde{T}$ as

$$R\tilde{T} = \frac{P_o}{R_o}\frac{p_o(\sigma)}{\rho_o(\sigma)}$$

then the approximation we make is that of dropping the order S_1 terms in the above relation. This is a reasonable approximation (see Browning and Kreiss, 1985).

We can now interpolate $\tilde{T}$ using the interpolation scheme (5.5). It can be easily shown that interpolating the dependent variables separately using (5.5) is equivalent to interpolating the geostrophic balance.

Tests with a model which uses the approximate equations and a mass conserving interpolation scheme verify that the vertically interpolated initial fields are balanced. For mass conservation the interpolation (5.5) must be linear. We also note that it is the geopotential (or pressure in other coordinate systems) which must be interpolated. This has dramatic consequences for the temperature fields. We discuss these consequences next.

5.3 Hydrostatic Balance

The mass field in a primitive equations model is completely defined by the surface pressure and the temperature. All other thermodynamic variables can be retrieved using the equation of state and the hydrostatic equation (3.3). The vertical variation of the mass field is completely defined by the temperature field. We can define a vertical interpolation of a thermodynamic variable based on the conservation of only one quantity because we have only one degree of freedom, we can vertically interpolate only one thermodynamic variable.

There are several quantities we may want to conserve when performing vertical interpolation of the thermodynamic variable. We also want to preserve the relative smoothness of the vertical profiles of all the thermodynamic variables. The following are important considerations when interpolating the thermodynamic variable.

1. The total potential energy is defined in equation (5.7). It can be shown (Lorenz, 1960) that only $\approx 0.5\%$ of this energy is available for conversion to kinetic energy and only $\approx 0.05\%$ is actually converted into kinetic energy. Conservation of TPE is important because small changes in TPE may lead to large changes in KE.
2. Vertical temperature profiles are observed to be smooth and do not generally exhibit folds except at the boundary between the troposphere and the stratosphere.
3. For large-scale flows the atmosphere is almost always statically stable. The convective adjustment scheme in the primitive equations model ensures this.

We have seen in Section 5.1 that in order to conserve mass we must interpolate the velocities linearly in the vertical direction. In Section 5.2 we showed that we needed first to introduce an approximation to the equations and then to interpolate the geopotential in the same manner in which we interpolated the velocities if we were to maintain the relative geostrophic balance in the atmosphere. Hence we must interpolate the geopotential linearly. We noted that this resulted in a different interpolation of the temperature which we defined with the matrix H.

Figure 28 shows three different interpolations of the geopotential (taken from the standard atmosphere) and the three resulting temperature profiles. Linear interpolation (in σ) of the geopotential results in an unphysical temperature profile. Linear

interpolation in $ln(\sigma)$ produces non-smooth temperature profiles. Only higher order interpolation in $ln(\sigma)$ will produce somewhat smooth temperature profiles.

The reason for this behavior is found by examining the discretization of the hydrostatic equation. The discretization of the hydrostatic equation (3.3) is

$$\phi_k - \phi_{k+1} = -\big(ln(\sigma_k) - ln(\sigma_{k+1})\big)\frac{R}{2}(T_k + T_{k+1}) \tag{5.10}$$

where increasing k corresponds to increasing σ and decreasing z. A physical interpretation of this hydrostatic equation is that the thickness of a layer ($\Delta\phi$) is determined by the layer's average temperature. We compute this temperature as the average of the temperature at its top and bottom. In this light one can see that the average temperatures produced by interpolating ϕ are physically reasonable but the resulting integration of the hydrostatic equation produces, in most cases, unreasonable oscillations about this average. This behavior can also be understood by examining the matrix A in (5.8). A is a rather poorly conditioned matrix and its inverse, used in computing H in (5.9), is not well behaved. Small changes in ϕ can result in large changes in T.

A higher order interpolation of ϕ in $ln(\sigma)$ smoothly interpolates $\partial\phi/\partial(ln\,\sigma)$ which leads to the significantly smoother temperature profiles. But even higher order interpolations of ϕ can lead to physically unrealistic temperature profiles.

The difficulties inherent in interpolating the geopotential and calculating the temperature field using the discretization (5.10) are well known. Initialization programs which prepare observational data for use in numerical integrations very carefully adjust the geopotential profiles to produce realistic temperature profiles. An example of a variational procedure which accomplishes this task along with a more detailed discussion of the problem of interpolating the geopotential can be found in Barker (1980). Many of these problems could be avoided by defining the temperature at the midpoints of the layers or possibly using higher order integration schemes for the hydrostatic equation. The popularity of the discretization (5.10) arises from the fact that using (5.10) along with a suitable discretization for the energy equation results in exact conservation of total potential energy.

Smooth vertical interpolation of the temperature does not ensure that the static stability of a column is maintained. Only if the potential temperature is interpolated

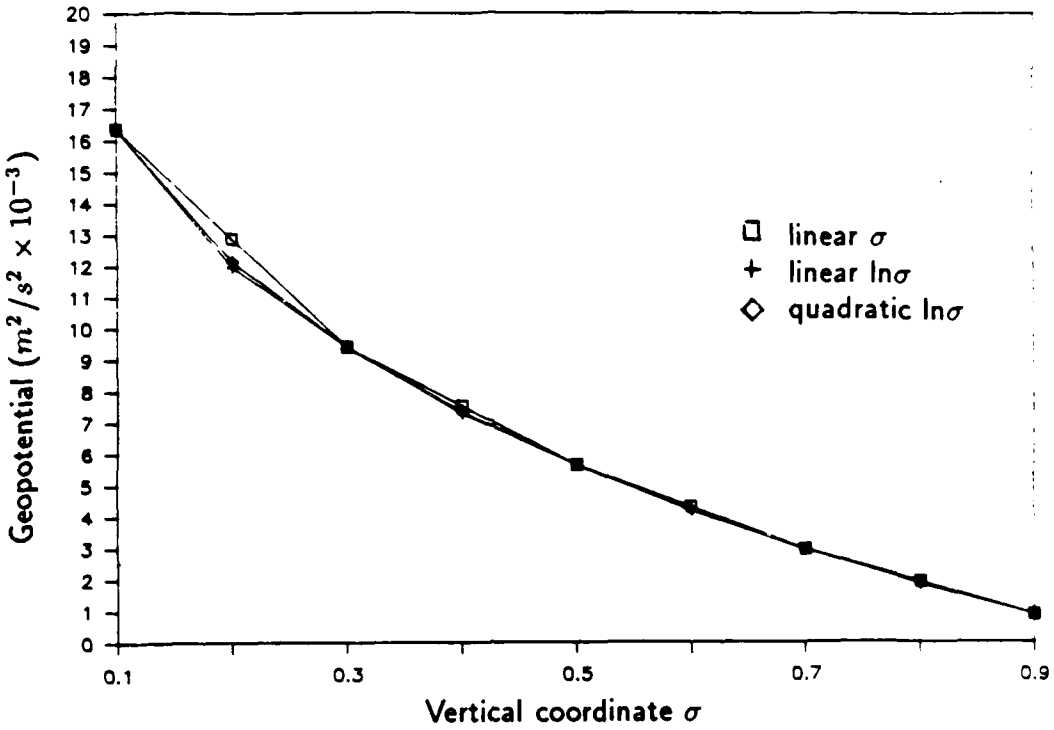


Figure 28A Geopotential using different interpolation methods.

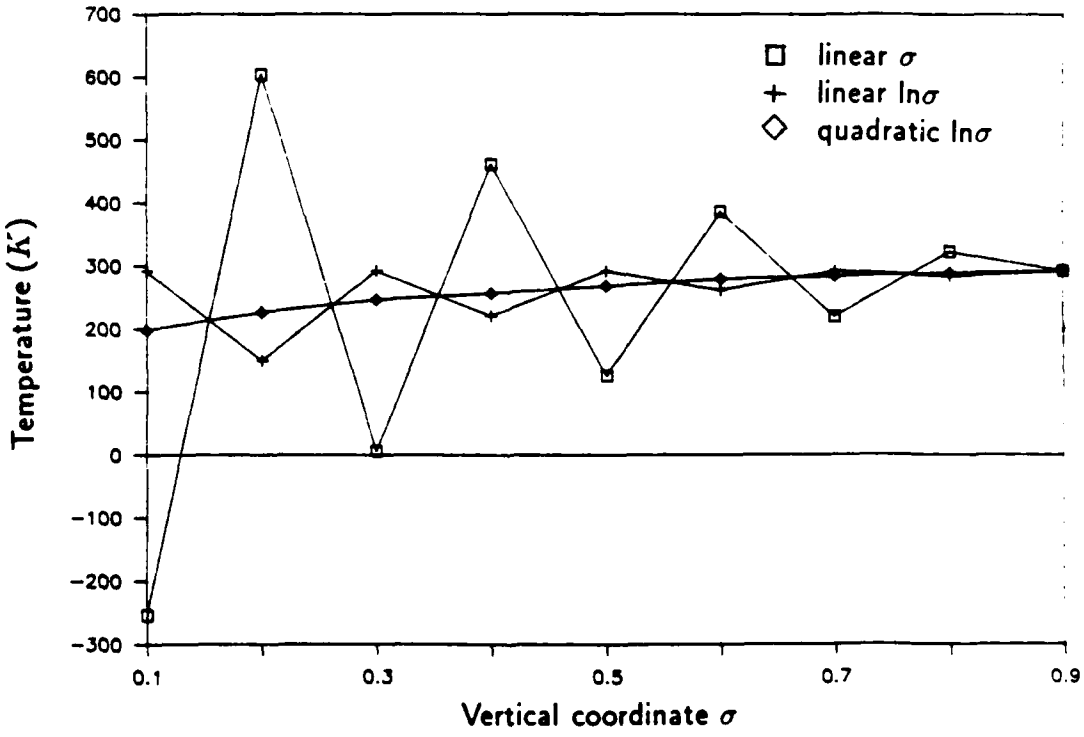


Figure 28B Temperatures derived from the hydrostatic equation and the geopotentials of Figure 28A.

is the stability profile preserved. Unstable profiles are immediately mixed by the convective adjustment routine described in Chapter 3. Interpolations which produce unstable profiles from an originally stable profile often lead to noisy and even unstable integrations.

We must finally consider the conservation of total potential energy. It is easily shown that TPE will be preserved in the current scheme if the temperature is linearly interpolated in the vertical. In tests we have found that when using a scheme which interpolates the temperature, and not the geopotential, we must interpolate the temperature linearly in the vertical and use an averaging scheme when updating (as is used for mass conservation) in order to conserve TPE. When interpolating the geopotential we do not update the temperature or the geopotential profiles on the coarse grid due to the difficulty of calculating the temperature from the geopotential. In these tests TPE was conserved on the coarse grid as long as mass was conserved.

5.4 Failure of the Primitive Equations

In Sections 5.1 through 5.3 we have outlined the constraints under which the vertical interpolation scheme is placed. To reiterate, these constraints are

1. the necessity that our interpolations conserve mass,
2. the necessity of interpolating the relative geostrophic balance (hence we must interpolate the geopotential in the manner in which we interpolate the velocity, i.e., we cannot directly interpolate the temperature) and
3. the necessity that the interpolations produce smooth temperature profiles and, of equal importance, that we do not create statically unstable regions through our interpolations.

The possible solutions we have outlined to these problems are not compatible. Choosing to satisfy some constraints automatically discounts the possiblity of satisfying the others. We have also found that using a different interpolation scheme for setting boundary values than that used in the initialization does not result in any cleaner or more stable solutions. No solution to these problems is obvious, especially if the current form and discretization of the equations is kept. Different discretizations and approximations may yield better results but it is unclear whether or not the mass and energy conserving nature of the current scheme can be preserved.

The most serious problem is centered around the need to vertically interpolate the geopotential coupled with the sensitivity of the temperature profile which is derived using the discretized hydrostatic equation (5.10) and the interpolated geopotential. Only temperature can be safely interpolated if the current discretization of the hydrostatic equation is to be used.

The situation might be characterized as one where there are too many constraints and not enough degrees of freedom. Thus, it appears that the primitive equations, in their current form and discretization, are not suited for use in a vertically refining model.

5.5 Theoretical Considerations for Vertical Refinement with the Primitive Equations

Oliger and Sundström (1978) have shown that the hydrostatic primitive equations are ill-posed for the initial boundary value problem with open boundaries. More specifically, they state that "local, pointwise boundary conditions cannot yield a well-posed problem for the open boundary problem using the hydrostatic (primitive) equations". A discrete approximation of a set of equations cannot have solutions which behave reasonably if it accurately approximates an ill-posed problem (Thomée, 1969). Yet the hydrostatic primitive equations have long been used as a basis for numerical weather prediction models which have performed well. In this section we examine the mechanism responsible for the ill-posedness as it is revealed in the analysis of Oliger and Sundström. We discuss the success of the primitive equations model, including the horizontally refining adaptive model, and comment on difficulties encountered using limited area models. We will also suggest that adaptive or nested models which employ vertically refined fine grids may face more serious problems in light of the mechanism responsible for the ill-posedness of the primititve equations.

The analysis of Oliger and Sundström (O&S) begins with the primitive equations posed in x, y and z coordinates.

$$\begin{aligned}
(\frac{\partial}{\partial t} + u \cdot \nabla)u_H + \alpha \nabla_H p + F_H &= 0 \\
\alpha \frac{\partial p}{\partial z} + g &= 0 \\
(\frac{\partial}{\partial t} + u \cdot \nabla)\alpha - \alpha \nabla \cdot u &= 0 \\
(\frac{\partial}{\partial t} + u \cdot \nabla)p + p\gamma \nabla \cdot u &= 0
\end{aligned} \tag{5.11}$$

In this set $u_H = (u_1, u_2)^T, u = (u_H, w)^T, F_H = (F_1, F_2)^T$ and $\nabla_H = (\partial/\partial x, \partial/\partial y)^T$.

The set cannot be analyzed with methods commonly used for hyperbolic systems of equations because the set is not hyperbolic. This can be shown by first writing the set in variational form and then examining the constant coefficient problem. Searching for eigensolutions of the form

$$u'_j = \hat{u}'_j exp\{i(\nu t + \omega_1 x_1 + \omega_2 x_2 + \omega_3 z)\}$$

reveals that the roots of the characteristic polynomial exhibit the asymptotic behavior

$$\tilde{\nu}_1 = O(1)$$

and

$$\tilde{\nu}_j = O(\frac{\omega_1}{\omega_3}) + O(\frac{\omega_2}{\omega_3}), \quad j = 2, 3$$

as $\omega_k \to \infty$ and $\tilde{\nu} = \nu + \sum_{j=1}^{2} u_j\omega_j + w\omega_3$. The signal speeds are inversely proportional to ω_3 and hence the system is not hyperbolic. General methods for analyzing hyperbolic systems, such as energy methods, cannot be used.

O&S perform the normal mode analysis upon the primitive equations. The variational set is linearized about an underlying basic state $\tilde{\alpha}$ and $\tilde{p}(z)$ which satisfies the hydrostatic equation $\tilde{\alpha}\tilde{p}_z + g = 0$. The variables can be rewritten as

$$\alpha = \tilde{\alpha}(z) + \alpha'$$

$$p = \tilde{p}(z) + p'$$

$$u_H = v + u'_H \quad v = \{v_1, v_2\}^t = const.$$

Substituting these into the set (5.11) and neglecting all nonlinear terms involving only primed quantities leads to an approximate system. The hydrostatic equation and the prognostic equations for p' and α' can be combined to yield the following system.

$$\frac{\partial}{\partial t}u'_H + \sum_{j=1}^{2} v_j \frac{\partial}{\partial x_j} u'_H + \nabla_H(\tilde{\alpha}p') + F' = 0$$

$$(\frac{\partial}{\partial t} + \sum_{j=1}^{2} v_j \frac{\partial}{\partial x_j}) L(\tilde{\alpha}p') + \nabla_H \cdot u'_H = 0$$

where

$$L(\tilde{\alpha}p') = -\frac{\tilde{\alpha}}{g^2}\frac{\partial}{\partial z}(\frac{\tilde{\alpha}}{\sigma}\frac{\partial}{\partial z}(\tilde{\alpha}p'))$$

and $\sigma = -g^{-1}\tilde{\alpha}^2(\partial/\partial z)ln\theta$ is a measure of the static stability of the basic state atmosphere. We assume $\sigma > 0$, i.e., we are studying perturbations to a statically stable atmosphere.

This system is separable and the variables u'_H and $\tilde{\alpha}p'$ can be expanded in the eigenfunctions F_ν of L. The system obtained is

$$(\frac{\partial}{\partial t} + \sum_{j=1}^{2} v_j \frac{\partial}{\partial x_j}) u'_{H_\nu} + \nabla_H(\tilde{\alpha}p')_{(\nu)} + F_{(\nu)} = 0,$$

$$(\frac{\partial}{\partial t} + \sum_{j=1}^{2} v_j \frac{\partial}{\partial x_j}) \lambda_\nu(\tilde{\alpha}p')_{(\nu)} + \nabla_H \cdot u'_{H_{(\nu)}} = 0$$

where the eigenvalues of L are λ_ν.

The eigenvalues λ_ν are distinct, positive real numbers and for each ν the system is hyperbolic. The characteristic velocities are v_j, $v_j - c_\nu$ and $v_j + c_\nu$, where $c_\nu = \lambda_\nu^{-\frac{1}{2}}$. Two classes of eigensolutions can be defined and they depend upon the value of c_ν relative to v_j.

$c_\nu > \lvert v_j \rvert$		$c_\nu < \lvert v_j \rvert$	
<u>inflow</u>	<u>outflow</u>	<u>inflow</u>	<u>outflow</u>
$v_j < 0$	$v_j > 0$	$v_j < 0$	$v_j > 0$
$v_j + c_\nu > 0$	$v_j + c_\nu > 0$	$v_j + c_\nu < 0$	$v_j + c_\nu > 0$
$v_j - c_\nu < 0$	$v_j - c_\nu < 0$	$v_j - c_\nu < 0$	$v_j - c_\nu > 0$

Proper specification of the boundary conditions requires that the number of boundary conditions specified equal the number of characteristics entering the region. The signs of the characteristic velocities determine whether the characteristics are entering or leaving the region and these signs depend upon the values of c_ν. Values of c_ν have been computed for a standard atmosphere with no mean wind shear by Wiin-Nielsen (1964). The values for the first sixteen modes are listed in Table 1. The first mode ($\nu = 0$) corresponds to an external gravity wave and is essentially the same as a free surface wave. The other modes are internal gravity waves.

v_j ranges from $O(1m/s)$ to $O(10m/s)$. For $|v_j| < c_\nu$ and $v_j < 0$ (inflow) only two variables should be prescribed at the boundary. This is likely to be the case for the first several modes. For large ν, $|v_j|$ is greater than c_ν and three variables should be prescribed at the inflow boundaries. Thus the number of variables which need to be prescribed depends upon the wavenumber ν. The number of variables which need specification at outflow boundaries is also dependent upon ν. The dependence of the number of variables needing specification upon the vertical wavenumber ν is the cause of the ill-posedness of the primitive equations. The only way to properly set local boundary conditions is to formulate these conditions in terms of local eigenfunction expansions or to use nonlocal boundary operators (O&S). No successful use of nonlocal boundary operators has been reported.

Next we need to answer the question of how this theoretical result applies to practical computations. While the equations are ill-posed for the open boundary

Table 1 Values of c_ν versus ν.
From Wiin Neilsen, 1964. (c_ν in m/s)

ν	c_ν	ν	c_ν
0	322.4	8	4.4
1	34.4	9	3.9
2	17.4	10	3.5
3	11.6	11	3.2
4	8.7	12	2.9
5	7.0	13	2.7
6	5.8	14	2.5
7	5.0	15	2.3

problem, they are weakly well-posed for problems without open boundaries. The success of global or hemispheric PE models is understandable for there are no open boundaries. Physical boundary conditions, such as the solid walls of a channel, also result in a weakly well-posed problem.

For limited area models and for nested models the problem is ill-posed. Limited area modellers have traditionally circumvented the problem of ill-posedness and resulting exponential error growth by including horizontal dissipation in their models and by using sponge-type boundary conditions along with increased horizontal dissipation close to the boundaries. This leaves open the question of exactly what equations are being solved and the accuracy of the limited area model solutions. The question of the effects of the ill-posedness of the primitive equations remains.

Problems with noise emanating from limited area model boundaries are the result of inaccurately specifying the boundary values. Filtering and viscosity may stabilize the boundary conditions but the quality of the solutions decays in time due to the ill-posed problem and inaccurate boundary values. Noise arises because there is no mechanism for preventing disturbances (regions of high solution error) from passing through the fine grid boundaries onto the coarse grid. Nested models that are not adaptive cannot control the solution error or the error at the solutions boundaries and hence are not reliable.

The question also arises as to whether or not the primitive equations are ill-posed

for solution on the local refinements (fine grids). This would be the case if we chose to treat a fine grid as a separate IVBP and use boundary conditions appropriate for the equations. Instead we have chosen to use continuity conditions and interpolate all data from one grid on to the boundary of another. Our interpolations are of sufficient accuracy so as not to introduce gross errors at the solution boundaries. This is achieved by always placing coarse-fine grid boundaries in regions where the errors are small. Problems concerning the ill-posedness of the equations do not arise.

In the discretized set of equations, the number of modes present in the solution corresponds to the number of vertical levels in the model. n-level model solutions contain the external mode ($\nu = 0$) and $n - 1$ internal modes. For non-vertically refining adaptive or nested models the same modes are present on the coarse and on the fine grids. This may be a reason for the success of the non-vertically refining adaptive model.

In a vertically refining model, the number of modes present in a fine grid solution will be greater than the number present in the coarse grid solution. These higher modes may develop on the fine grid and, given they are not present on the coarse grid, appear as sources of error in the setting of boundary values for the fine grid. This source of error may prove particularly troublesome when a modal breakdown of the conditions indicates that the number of entering or leaving characteristics differ on the coarse and fine grids.

This last problem is very likely to occur in practice. For example, if we have a coarse grid with less than eight layers, a nested fine grid or adaptive fine grid with more than eight layers and a mean wind of 5 meters per second, then at the outflow boundaries on the coarse grid it would be proper to specify two boundary conditions whereas on the fine grid it would be correct to specify two outflow conditions for the first 8 modes and only one boundary condition for the remaining modes. Present large-scale atmospheric models typically contain from 5 to 15 vertical layers and hence these problems can be anticipated if the models are used as the basis for nested or adaptive vertically refining models.

Wiin-Nielsen has also calculated values of c_ν for atmospheres with mean vertical wind shear. Table 2 lists some of these results. When vertical shear is present, the velocities c_ν increase. The crossover point, where c_ν changes from being greater than

Mean vertical shear (m/s/km)

ν	1.0	2.0	3.0	4.0
0	322.9	323.6	324.6	325.8
1	35.4	37.6	41.4	46.9
2	19.1	24.3	31.6	40.5
3	14.3	21.2	30.3	40.0
4	12.0	20.4	30.0	40.0
5	11.1	20.1	30.0	40.0
6	10.7	20.0	30.0	40.0

Table 2 c_ν versus ν.
with mean vertical wind shear.
From Wiin Neilsen, 1964. (c_ν in m/s)

v_j to being less than v_j, now occurs at a higher wavenumber ν. For a given wind speed and wind shear the crossover point could be calculated. For low shear it is still likely that the coarse grid may not contain the crossover wave while the fine grid will. For higher levels of shear it becomes unlikely that wind speeds will reach levels where $c_\nu < v_j$ in which case there is no crossover.

We have not been able to construct a vertically refining model which would allow us to explore the ill-posedness of the boundary conditions and we have no computational example of the problems we have outlined in this section. We believe that the ill-posed boundary conditions would adversely affect solutions for vertically refined nested or adaptive models and that a well-posed set of equations is needed as a basis for a vertically refining model.

6 Browning-Kreiss Equations

The primitive equations are used in large-scale atmospheric flow models for two major reasons. First, the hydrostatic approximation removes sound waves from the solution. Large time steps can be taken because vertically propagating sound waves no longer restrict the size of the time step through a limiting CFL condition. Second, sigma coordinates can be used resulting in greatly simplified bottom boundary conditions. The primitive equations possess several disadvantages; they are ill-posed for the open boundary value problem, appear intransigent to use in a vertically refining model, and are limited to use for large-scale flows.

We wish to use an equation set which, while not sacrificing the advantages of the primitive equations, removes the difficulties associated with their use. In this chapter we present the Browning-Kreiss (BK) equations. This equation set constitutes a well-posed system for the open boundary problem. It is a non-hydrostatic set and, although derived for large-scale motions, permits the computation of non-hydrostatic motions on smaller scales. The new set also can be economically integrated in large-scale flow models. Unfortunately, interpolation problems remain which hinder the development of a vertically refining model. These problems can be linked to the approximate hydrostatic balance which exists in the atmosphere.

Presented in the first section of this chapter is a brief derivation of the BK equations. A fully explicit finite difference model has been constructed for solving these equations and it is described in the second section. The last section of this chapter outlines the vertical interpolation problems. We have not been able to construct a vertically refining model with the BK equations.

6.1 Derivation

The derivation of the BK equations relies upon the use of the bounded derivative method. The method was developed by Kreiss (1980) to prepare initial data for symmetric hyperbolic systems posessing multiple time scales. For large-scale atmospheric motions, fast moving gravity waves and sound waves are not of meteorological interest and appear only as noise in the solutions while motions of interest, such as Rossby waves, evolve on slower time scales. The bounded derivative method is used to develop a set of equations which removes the effects of the fast moving waves.

The fundamental concept of the method, as used in the initialization process, is simple: Choose the initial data in such a way that at $t = 0$ a number $k > 0$ of time derivatives are of order unity. Using this principle, constraints consisting of partial differential equations are derived which the initial data must satisfy. The number of linearly independent constraints does not depend on k, but rather the constraints become more refined with increasing k. The method results in solutions which only vary on a slow time scale over some time interval with an upper bound of T. The size of T depends on k. The larger the number of derivatives of order unity, the longer it will take for fast waves to appear in the solution.

Browning and Kreiss (1986) use this method to derive the BK equations. The constraints which they impose on the initial data in order to suppress fast moving, short waves are now used to derive a reduced system of equations. The bounded derivative principle as applied to the derivation of a reduced system is: If the solution of a system of equations varies slowly with respect to time, then it must have a number of nondimensional time derivatives of the order unity. It is required that the time derivatives vary slowly not only at $t = 0$, but throughout all time. The following is a brief derivation of the BK equations and follows the derivation of Browning and Kreiss (1985, 1986).

The derivation begins with a scaling of the inviscid Euler equations appropriate for large-scale atmospheric flow. We can write the Euler equations in the x, y, z and t coordinate system as

$$\frac{ds}{dt} = 0 \tag{6.1a}$$

$$\frac{dp}{dt} + \gamma p \nabla \cdot V = 0 \tag{6.1b}$$

$$\frac{dV}{dt} + \frac{1}{\rho}\nabla p + f(k \times V) + gk = 0 \tag{6.1c}$$

where p is the pressure, $V = (u, v, w)^T$ is the transposed velocity vector, ρ the density, $s = \rho p^{-1/\gamma}$, $f = f(y)$ is the Coriolis parameter, g the gravitational constant, γ the adiabatic exponent ($\gamma = 1.4$), k the vertical unit vector and

$$\frac{d}{dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} + w\frac{\partial}{\partial z}.$$

This set is scaled and nondimensionalized by introducing dimensionless variables which are assumed to be $O(1)$. The dimensionless variables, denoted by primes, are

$$\begin{aligned} x &= L_1 x' & u &= U u' \\ y &= L_2 y' & v &= V v' \\ z &= D z' & w &= W w' \\ t &= T t' \end{aligned}$$

$$p = P_o[p_o(z) + S_1 p']$$
$$\rho = R_o[\rho_o(z) + S_1 \rho'].$$

Here, $p_o(z)$ and $\rho_o(z)$, the horizontal means of pressure and density, satisfy the hydrostatic relationship

$$P_o \frac{dp_o}{dz} + g R_o \rho_o(z) = 0.$$

P_o and R_o are typical mean surface values of the pressure and density. S_1 is chosen so that ρ' and p' are O(1) with the result being that $10^{-2} < S_1 < 10^{-1}$. The magnitude of S_1 represents the observation that the deviations of the pressure and the density from the horizontal means ($p_o(z)$ and $\rho_o(z)$) are small. s is proportional to the reciprical of the potential temperature and can be written as

$$s = R_o P_o^{-1/\gamma} s_o(z)(1 + S_1 s')$$

with

$$s_o(z) = \rho_o(z) p_o(z)^{-1/\gamma}$$

and

$$s' = \frac{\rho'}{\rho_o} - \frac{1}{\gamma}\frac{p'}{p_o} + O(S_1).$$

g and f can be nondimensionalized with

$$g = Gg'$$

$$f = 2\Omega[f_o + \frac{L_2^2}{r}\beta y']$$

where $G = 10\text{m/s}^2$ and r is the mean radius of the earth.

The scaling parameters have been chosen so that all variables and constants in the original equations (6.1) are dimensionless and O(1). By substituting the dimensionless variables into the equations, defining a new variable $\phi' = p'/\rho_o(z)$, arranging the equations such that the time derivatives are $O(1)$ and dropping terms of $O(S_1)$ or smaller we obtain the scaled system given by

$$\frac{ds'}{dt'} + (10S_1)^{-1}\tilde{S}_2\tilde{s}(z)w' = 0 \tag{6.2a}$$

$$\frac{d\phi'}{dt'} + \tilde{S}_2\tilde{\rho}\phi'w' + S_1^{-1}p_o\rho_o^{-1}[\gamma d + \tilde{S}_2\tilde{p}(z)w'] = 0 \tag{6.2b}$$

$$\frac{du'}{dt'} + S_3\phi'_{x'} - S_4 f'v' = 0 \tag{6.2c}$$

$$\frac{dv'}{dt'} + \tilde{S}_3\phi'_{y'} + \tilde{S}_4 f'u' = 0 \tag{6.2d}$$

$$\frac{dw'}{dt'} + S_1S_5[\phi'_{z'} + S_6(0.1\tilde{s}\phi' + g's')] = 0, \tag{6.2e}$$

where

$$\frac{d}{dt'} = \frac{\partial}{\partial t'} + u'\frac{\partial}{\partial x'} + v'\frac{\partial}{\partial y'} + S_2 w'\frac{\partial}{\partial z'}$$

$$d = u'_{x'} + v'_{y'} + S_2 w'_{z'}$$

$$\tilde{p}(z) = D_o(lnp_o)_z$$

$$\tilde{s}(z) = 10D_o(lns_o)_z$$

$$\tilde{\rho}(z) = D_o(ln\rho_o)_z.$$

The quantities $\tilde{p}_o(z)$, $\tilde{s}_o(z)$ and $\tilde{\rho}_o(z)$ are defined such that they are O(1). D_o is of the order of the largest equivalent depth of the atmosphere (the external mode), and $D \leq D_o$. The factor of ten in the definition of $\tilde{s}_o(z)$ is to insure that $\tilde{s}_o(z)$ is O(1). The remaining dimensionless parameters are defined by

$$S_2 = D^{-1}TW\ , \qquad \tilde{S}_2 = D_o^{-1}TW,$$

$$S_3 = S_1 P_o (R_o U^2)^{-1}, \qquad \tilde{S}_3 = S_1 P_o (R_o V^2)^{-1}$$
$$S_4 = 2\Omega T L_1^{-1} L_2, \qquad \tilde{S}_4 = 2\Omega T L_1 L_2^{-1},$$
$$S_5 = T P_o (D R_o W)^{-1},$$
$$S_6 = D G P_o^{-1} R_o = D D_o^{-1}.$$

Using a scaling analysis based upon this nondimensionalization, Browning and Kreiss derive a reduced set of equations which accurately describe the dynamics of many scales of atmospheric motion. We do not present the scaling arguments here, rather, the reader should consult the reference (Browning and Kreiss, 1985,1986). The system which describes adiabatic, smooth stratified flow is

$$\frac{ds'}{dt'} + \tilde{s} w' = 0 \tag{6.3}$$
$$\epsilon^{3-n} \frac{d\phi'}{dt'} + p_o \rho_o^{-1} [\gamma d + \epsilon^{2-n} \tilde{p} w'] = 0 \tag{6.4}$$
$$\epsilon^n \frac{du'}{dt'} + \phi'_{x'} - f v' = 0 \tag{6.5}$$
$$\epsilon^n \frac{dv'}{dt'} + \phi'_{y'} + f u' = 0 \tag{6.6}$$
$$\eta \frac{dw'}{dt'} + \phi'_{z'} + 0.1 \tilde{s} \phi' + g' s' = 0 \tag{6.7}$$

where all variables are nondimensional, of O(1) and

$$\frac{d}{dt'} = \frac{\partial}{\partial t'} + u' \frac{\partial}{\partial x'} + v' \frac{\partial}{\partial y'} + \epsilon^{2-n} w' \frac{\partial}{\partial z'}$$
$$d = u'_{x'} + v'_{y'} + \epsilon^{2-n} w'_{z'}.$$

The parameters for the reduced set are η, ϵ and n. They are defined as

$$\epsilon^{-n} = S_3 = S_4,$$

$$\eta^{-1} = S_1 S_5$$

with $\epsilon = 10^{-1}$, $n \geq 1$ and $0 < \eta < 10^{-4} \epsilon^{2n}$. The values of the scaling parameters η, ϵ and n are determined by the scales present in the flow and the initial scaling requirement that all terms other than the scaling parameters be O(1). Various scalings are discussed by Browning and Kreiss (1985,1986). For large-scale atmospheric flows with length scales $L_1 = L_2 = L = 10^6$ meters and $D = 10^4$ meters, time scale $T = 10^5$ seconds, and velocity scales $U = V = 10$ meters per second and $W = 10^{-2}$

meters per second, Browning and Kreiss show that the scaled equations are consistent and that the parameters η and n have the values 10^{-6}, and 1 respectively.

From this set many simpler sets of equations can be derived which correspond to the classic sets used in numerical weather prediction. The primitive equations can be derived from 6.3–6.7 by setting $\eta = 0$, i.e., by removing the material derivative of w'. This approximation removes sound waves from the solution and also replaces the vertical momentum equation with the hydrostatic equation. In a physical, intuitive sense, we can view setting $\eta = 0$ as equivalent to increasing the vertical sound-wave speed in the equations to infinity. The approximation is accurate but leaves the system ill-posed. A different approximation is needed.

Browning and Kreiss replace equation 6.7 by

$$\alpha^{-1}\eta\frac{dw'}{dt'} + \phi'_{z'} + 0.1\tilde{s}\phi' + g's' = 0, \tag{6.8}$$

where $\alpha \leq 1$.

The bounded derivative method is used to determine α. The method requires that the solution of the new system of equations must be smooth, hence, it must have time derivatives of order unity. The equations have been scaled such that the first derivatives in time are O(1) or smaller, thus, second derivatives in time must be examined.

The second derivative in time of the scaled pressure ϕ' can be isolated by rewriting equation 6.4 as

$$\epsilon^n(u'_{x'} + v'_{y'}) + \epsilon^2(w'_{z'} + \gamma^{-1}\tilde{p}w') = O(\epsilon^3).$$

We can take the time derivative of this equation and, using equations 6.3, 6.5, 6.6 and 6.8 to replace the time derivatives with space derivatives, arrive at the elliptic equation for ϕ'

$$\begin{aligned}&\phi'_{x'x'} + \phi'_{y'y'} + \alpha\epsilon^2\eta^{-1}[\phi'_{z'z'} + (\gamma^{-1}\tilde{p} + 0.1\tilde{s})\phi'_{z'} + 0.1(\gamma^{-1}\tilde{p}\tilde{s} + \tilde{s}_{z'})\phi'] = \\ &f\zeta - \alpha\epsilon^2\eta^{-1}g(s'_{z'} + \gamma^{-1}\tilde{p}s') + 2\epsilon^n(u'_{z'}v'_{y'} - u'_{y'}v'_{z'}) + O(\epsilon^2).\end{aligned} \tag{6.9}$$

Smooth solutions to the set 6.3 – 6.6, 6.8 exist only if the coefficients in 6.9 are of order unity. Consequently, α is chosen so as to satisfy this requirement, hence

$$\alpha = \epsilon^{-2}\eta.$$

It can also be shown (see Browning and Kreiss, 1985, 1986) that

$$\alpha = \left(\frac{D}{L}\right)^2.$$

Large-scale atmospheric motions with the length, time and velocity scales listed in this section lead to the value $\alpha = 10^{-4}$.

Physically, this approach is opposite to that used in deriving the primitive equations. The speed of vertically propagating sound waves in the new set of equations is *decreased* by choosing a value of $\alpha \leq 1$. It can be shown, using the standard linear analysis, that the new sound wave speed is $\sqrt{\alpha} \cdot \sqrt{\gamma RT}$ (for large-scale flows, $L >> D$) whereas the actual speed of sound is $\sqrt{\gamma RT}$. This is in contrast to the primitive hydrostatic equations ($\eta = 0$) where the speed of the vertically propagating sound waves is *increased* to infinity, i.e., the sound waves no longer exist.

The introduction of α also effects other wave motions. For example, it can be shown that the gravity waves present in hydrostatic motions propagate at only half their true velocity when using the Browning-Kreiss equations. The critical point is that the use of α introduces only small errors in motions for which the equations have been scaled. Motions smaller than those for which the equations have been scaled are unimportant in the calculation of the larger scales and, in any case, cannot be adequately resolved in computations.

Browning and Kreiss show that the system 6.3–6.6 and 6.8 forms a well-posed set for the open boundary value problem. The system can be used to simulate several scales of atmospheric motion. These motions include

1. hydrostatic motions with equal horizontal length scales,
2. hydrostatic motions with unequal horizontal length scales,
3. non-hydrostatic motions with equal horizontal length scales,
4. non-hydrostatic motions with unequal horizontal length scales and
5. diabatic motions.

6.2 Browning-Kreiss Solver

We have developed a solver for the BK equations (6.3–6.6, 6.8). The solver is tested by simulating a developing baroclinic disturbance in a periodic channel. For this flow the scaling parameters in the equations are

$$U = 10 \text{ m/s}, \qquad L = 10^6 \text{ meters},$$
$$V = 10 \text{ m/s}, \qquad D = 10^4 \text{ meters},$$
$$W = 10^{-2} \text{ m/s}, \qquad T = 10^5 \text{ seconds},$$

and the dimensionless parameters are

$$\alpha = 10^{-4}, \qquad \eta = 10^{-6}$$
$$\epsilon = 10^{-1}, \qquad n = 1.$$

The equations contain several parameters which are a function of z and are defined in terms of mean thermodynamic variables. The standard atmosphere is used to compute these functions and the results are listed in Table 3. Note that all the dimensionless parameters are $O(1)$. We drop the primes in the rest of this work and assume that all variables are dimensionless unless otherwise noted.

The horizontal discretization is accomplished on the C–grid, the same grid used in the primitive equations solver described in Chapter 3. The horizontal velocities u and v are staggered with respect to s and ϕ. This discretization is used in the primitive equations solver and is illustrated in Figure 6; s and ϕ are carried at the p points in the figure. The vertical discretization is shown in Figure 29. The vertical velocities are defined on horizontal planes lying between the planes where u, v, s and ϕ are defined. The vertical discretization is similar to the discretization used in the primitive equations model; however, the vertical coordinate in this model is the geometric height z and not the normalized pressure p/p_s. The vertical velocity w is located directly above (or below) the s and ϕ points on the C–grid, halfway between layers of the C–grid, as depicted in Figure 29.

The equations are discretized with second-order centered-in-time, centered-in-space differencing. The scheme is fully explicit. Details of the discretization are contained in Appendix 3.

The BK equations are in advective form whereas are the primitive equations are in flux form. Thus, boundary conditions, particularly the vertical boundary conditions,

height (km.)	pressure (pascals)	density (kg/m^3)	$p_o(z)$	$\rho_o(z)$	$s_o(z)$	$\tilde{s}(z)$	$\tilde{p}(z)$
32.00	513.	0.0130	0.0051	0.0130	0.5622	-0.3127	-2.2450
28.00	1258.	0.0250	0.0126	0.0250	0.5692	-1.1232	-2.0542
24.00	2651.	0.0460	0.0265	0.0460	0.6150	-2.9239	-1.7930
20.00	5280.	0.0880	0.0528	0.0880	0.7192	-4.0113	-1.6951
18.00	7330.	0.1210	0.0733	0.1210	0.7824	-4.0698	-1.6304
16.00	10136.	0.1650	0.1014	0.1650	0.8464	-4.1985	-1.6143
14.00	13982.	0.2270	0.1398	0.2270	0.9254	-4.3838	-1.6047
12.00	19259.	0.3110	0.1926	0.3110	1.0086	-3.5840	-1.5845
10.00	26352.	0.4120	0.2635	0.4120	1.0681	-2.4104	-1.5497
9.00	30659.	0.4660	0.3066	0.4660	1.0843	-1.4568	-1.4927
8.00	35520.	0.5250	0.3552	0.5250	1.0996	-1.4262	-1.4519
7.00	40989.	0.5900	0.4099	0.5900	1.1156	-1.3488	-1.4131
6.00	47120.	0.6600	0.4712	0.6600	1.1297	-1.2310	-1.3754
5.00	53967.	0.7360	0.5397	0.7360	1.1434	-1.2251	-1.3394
4.00	61595.	0.8190	0.6159	0.8190	1.1577	-1.2299	-1.3056
3.00	70070.	0.9090	0.7007	0.9090	1.1719	-1.2328	-1.2739
2.00	79468.	1.0070	0.7947	1.0070	1.1866	-1.1937	-1.2439
1.00	89862.	1.1120	0.8986	1.1120	1.2002	-1.1206	-1.2149
0.00	101325.	1.2250	1.0132	1.2250	1.2135	-1.1025	-1.2006

Table 3 Standard atmosphere parameters for the Browning-Kreiss model.

differ.

For flow in a channel we assume that all solid boundaries are free slip surfaces with no normal fluxes. Thus, at the North and South channel walls ($y = Y_S, Y_N$) we impose the conditions

$$v|_{y=Y_S,Y_N} = 0$$

and

$$\left.\frac{\partial}{\partial y}\right|_{y=Y_S,Y_N} (u, \phi, s, w) = 0.$$

The East-West boundaries are periodic.

In the primitive equations model the vertical velocity in the σ–coordinate system, $\dot{\sigma}$, is zero at the surface. This boundary condition is all that is required in the numerical

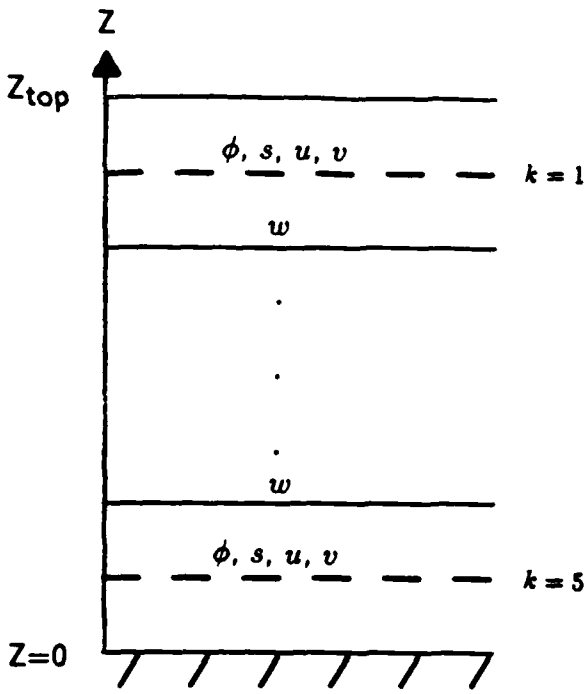


Figure 29 Vertical Discretization for the Browning-Kreiss solver.

scheme developed for a set of equations in flux form. For equations in advective form other conditions must supplement this condition. The boundary conditions at the surface for the BK solver are

$$w|_{z=0} = 0,$$

$$\left.\frac{\partial}{\partial z}\right|_{z=0} (u, v, s) = 0$$

and

$$\left(\frac{\partial \phi}{\partial z} + 0.1\tilde{s} + gs\right)\Big|_{z=0} = 0$$

This last condition is the hydrostatic constraint. At the bottom boundary this constraint is consistent because both w and dw/dt must be zero at the surface.

The upper boundary conditions present problems. The upper surface in our channel is located at a constant height Z_T. Since no mass, momentum or energy flux should occur through this boundary, it would appear that the correct boundary condition is $w = 0$ at $z = Z_T$. Using this condition with any other combination of conditions for u, v, s and ϕ results in unstable integrations. In the advective equations the vertical flux terms are all of the form $w\partial A/\partial z$; consequently, another possible form for the boundary conditions would be to specify that the vertical derivatives at the top are zero but not that the vertical velocity is zero. This formulation has proven

successful as is shown by the results in Chapter 7. The boundary conditions at the upper boundary $z = Z_T$ are

$$\left.\frac{\partial}{\partial z}\right|_{Z_T} (u, v, w, s, p) = 0.$$

The implementation of these boundary conditions in the discretized scheme, can be found in Appendix 3.

6.3 Vertical Refinement with the Browning-Kreiss Equations

In Chapter 5 we discussed the problems encountered when constructing a vertically refining model with the hydrostatic primitive equations. The critical problem, which we have yet to solve, is to interpolate the geopotential in a way which will produce smooth, realistic, statically stable temperature profiles. The temperature can be derived from the geopotential using the hydrostatic equation, and vice-versa.

The BK equations are not hydrostatic, hence, the temperature and geopotential, or in the BK equations the dimensionless perturbation pressure ϕ and the dimensionless perturbation s' to the reciprical of the potential temperature s, are no longer directly related through a hydrostatic equation. Each variable can be interpolated independently.

Use of the BK equations does not solve the interpolation problem. The reasons are:

1. For large-scale atmospheric flows the atmosphere is very nearly hydrostatic. The thermodynamic variables must be interpolated in a manner which preserves this near-hydrostatic balance, even though the BK equations do not use the hydrostatic approximation.
2. Problems similar to those described in Chapter 5 dealing with the interpolation of the geopotential and temperature are encountered when attempting to interpolate the thermodynamic variables ϕ and s so as not to disturb the near-hydrostatic balance.

These problems can be understood by examining the vertical momentum equation (6.8), which we repeat here in a different form,

$$\alpha^{-1}\eta\frac{dw}{dt} = -\phi_z - 0.1\tilde{s}\phi - gs. \tag{6.10}$$

All variables and constants are O(1) except for $\alpha^{-1}\eta = 10^{-2}$. Thus the RHS of (6.10) must be O(10^{-2}) if dw/dt is to be O(1). The RHS of (6.10) represents the departure of the atmosphere from a hydrostatic balance. Since the variables and constants are all O(1) we see that the RHS terms nearly cancel each other. In exchanging the primitive equations for the BK equations we have discarded the strict hydrostatic constraint and replaced it with a vertical momentum equation, but still, the atmosphere must be very nearly hydrostatic. The two thermodynamic variables in the BK equations cannot be interpolated independently because abnormally large vertical accelerations arise and the computations often prove unstable. What is needed is a vertical interpolation scheme for ϕ and s which will preserve the near-hydrostatic balance.

The normalized, nondimensional perturbation pressure ϕ must be interpolated in the same manner as u and v in order to preserve the near-geostrophic balance in the BK equations. One possible scheme for interpolating s is to interpolate the RHS of (6.10), i.e., interpolate the departure of the atmosphere from a hydrostatic balance, and then compute s. This scheme ensures that the near-geostrophic and near-hydrostatic balance is preserved and that the vertical accelerations are O(1). Unfortunately it also results in unrealistic profiles of the potential temperature. Unrealistic buckling of the s field occurs and statically unstable layers appear where none existed. This result is similar to the buckling of the temperature fields discussed in section 5.3 and illustrated in Figure 28B.

The problem with the vertical interpolation of s is the same problem found in the vertical interpolation of T. It is the *average* of s computed in the discretization of the RHS of (6.10) which must balance the *gradient* of ϕ. We have not found a solution to this problem.

7. TEST CASE RESULTS FOR THE BROWNING-KREISS EQUATIONS

In this chapter we present results from the adaptive Browning-Kreiss model. With these results we demonstrate that the BK equations correctly simulate large-scale, hydrostatic flow and that the adaptive components of the model perform properly. We conclude the chapter with a section which discusses the use of these equations and the adaptive model for simulating non-hydrostatic flows.

7.1 Baroclinically Unstable Jet

We have used the BK model to simulate the evolution of a baroclinically unstable jet which is subjected to an initial perturbation. The results described here are similar to those in Section 2 of Chapter 4. The two simulations cannot be directly compared because different coordinate systems, variables, initializations and equation sets are used. We consider it sufficient that the developing flow possess the features of a developing baroclinic disturbance, and given this, we conclude that the BK equations adequately describe large-scale atmospheric flow.

The initial conditions for the unstable jet are shown in Figure 30. The plots are in terms of the normalized velocity u and the perturbation variables ϕ and s. The channel width (N-S) is 12,000 kilometers and its length is 5040 kilometers. The beta-plane approximation is used. The initial conditions are derived by requiring that the fields be in geostrophic and hydrostatic balance. The vertical velocity is initially zero. The north-south velocity v is perturbed, thereby providing the perturbation to the jet. The perturbation is not balanced.

After several days simulated time the disturbance has developed. Figures 31, 32 and 33 are plots of the velocity vectors, pressure and potential temperature for the plane at $Z = 1$km. These plots show several of the prominent features of the developing baroclinic disturbance.

Figure 31 shows the cyclonic and anticyclonic circulations about the high and low pressure regions shown in Figure 32. These features can be compared with the wind and surface pressure patterns from the primitive equations model plotted in Figures 15 and 16. Although the strengths of the disturbances differ, the general features are present in both. Figure 33 is a plot of the potential temperature for

the BK model and Figure 14 is a comparable plot for the primitive equations model. Again we see that the fronts are similar in appearance. As in the primitive equation simulation, the warm and cold fronts at the lowest levels are located in regions of horizontal deformation and horizontal shear. Wind shifts are found at the fronts.

Figure 34 is a plot of the North-South velocity v on the vertical cross-section along the line AB in Figure 32. The location of the troughs and ridges are highlighted in Figure 34. We observe that the troughs and ridges tilt towards the west. This tilting is needed if the mean flow is to give up potential energy to the developing disturbance. We also observed that the low level, low and high pressure regions (troughs and ridges close to the surface) develop between the upper level troughs and ridges and this can also be observed in the plots. In the mature phase of the disturbance the troughs and ridges at the lower and upper levels coincide.

These results indicate that the BK equations are sufficient for use in the test case and that the solver is adequately accurate. More rigorous testing of the equations has been performed by Browning and Kreiss (1985, 1986).

We have performed adaptive calculations with the BK model using the fields shown in Figures 31 through 34 as initial conditions. The same principles are used for grid interaction in the adaptive BK equations model as are used in the adaptive primitive equations model. We wish to show that the fine-coarse grid interaction is correct. This interaction takes place during the setting of fine grid boundary conditions and updating of the coarse grid solution. We also want to show that the interaction between overlapping fine grids is correct. The solutions in the overlap region should be the same.

Figures 35 and 36 show the coarse grid pressure and potential temperature fields on the $Z = 1$km. surface after 36 hours simulation time for a run which used the grid setup shown in Figure 37. As in the primitive equations simulation, the maximum errors occur at the jet core and fine grids are placed over the jet. After 36 hours time the lower level low and high pressure regions have moved east with the low intensifying by a few millibars. This is not a strong disturbance but the simulation does illustrate that the interactions between the coarse and fine grids are correct from the coarse grid perspective. The locations of the fine grids is not apparent in the solutions. No noise arises from the updating procedure.

Figures 38 and 39 contain plots of the pressure and potential temperature on the fine grids in figure 37 at the $Z = 1$km. surface. In the overlap regions the solutions agree. There are no kinks in the fields at the boundaries. The fine grid boundary conditions are performing properly. There is significantly more detail in the fine grid cold and warm fronts than that which appears on the coarse grid. The coarse grid does appear to represent the pressure field adequately. A solution calculated on the coarse grid alone does not exhibit as much growth in the strength of the low pressure system and also diffuses the cold and warm fronts.

These results show that the adaptive BK model works correctly, at least in a qualitative sense. While both the adaptive BK model and the adaptive primitive equations model produce acceptable solutions for our test problem, there are advantages to using the BK model. The BK equations are well-posed for the open boundary problem. If the coarse grid has open boundaries, such as in a limited area model, correct boundary conditions can be used and excessive filtering and noise at the boundaries can be eliminated. We have also found that significantly less artificial viscosity is needed in the BK model as compared to the primitive equations model to avoid nonlinear instability. For example, in fully explicit calculations with the primitive equations model and a damping term for the quantity ϕ of the form

$$K\left(\frac{\partial^4\phi}{\partial x^4} + \frac{\partial^4\phi}{\partial y^4}\right)$$

we use

$$K = -\frac{1}{1280}\frac{\Delta x^4}{\Delta t}$$

whereas for the Browning-Kreiss model we use

$$K = -\frac{1}{5120}\frac{\Delta x^4}{\Delta t}.$$

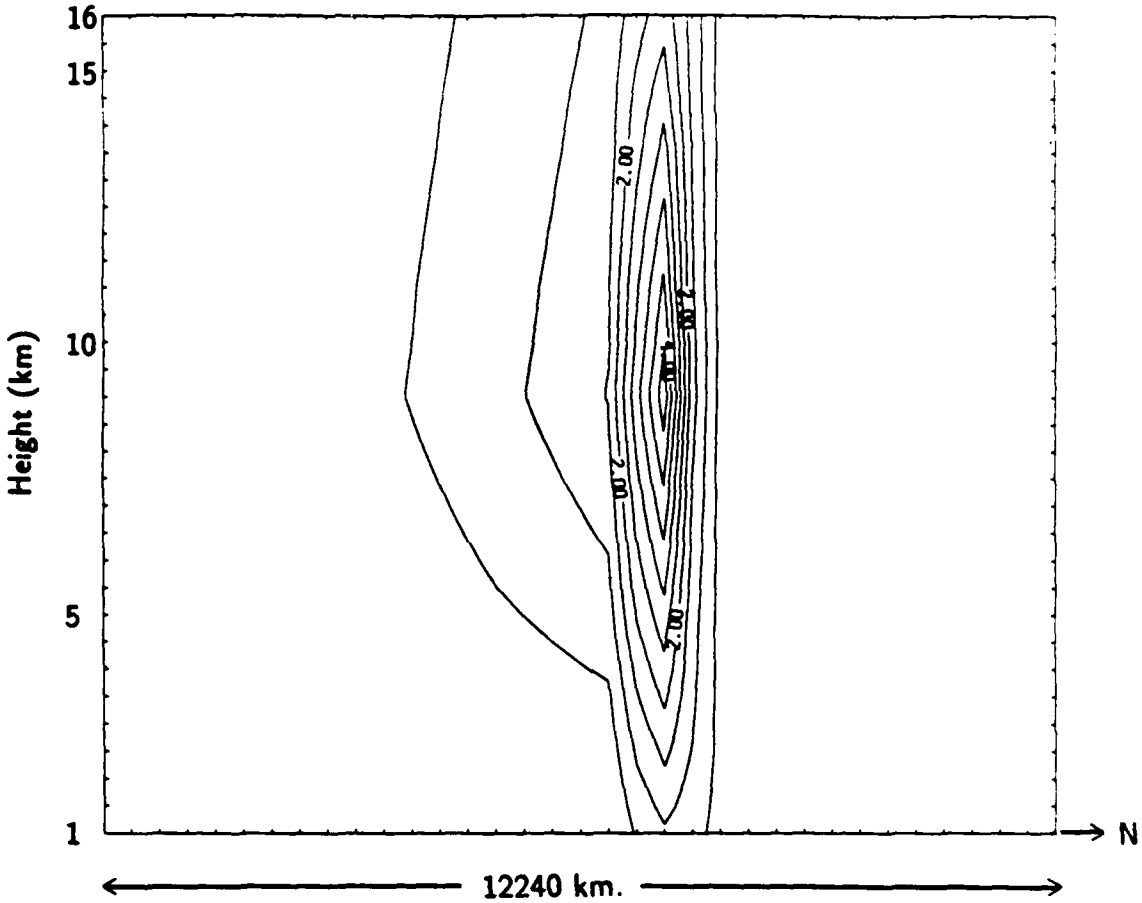


Figure 30A Initial jet. u' velocity (dimensionless) $u = Uu'$, $U = 10$m/s.

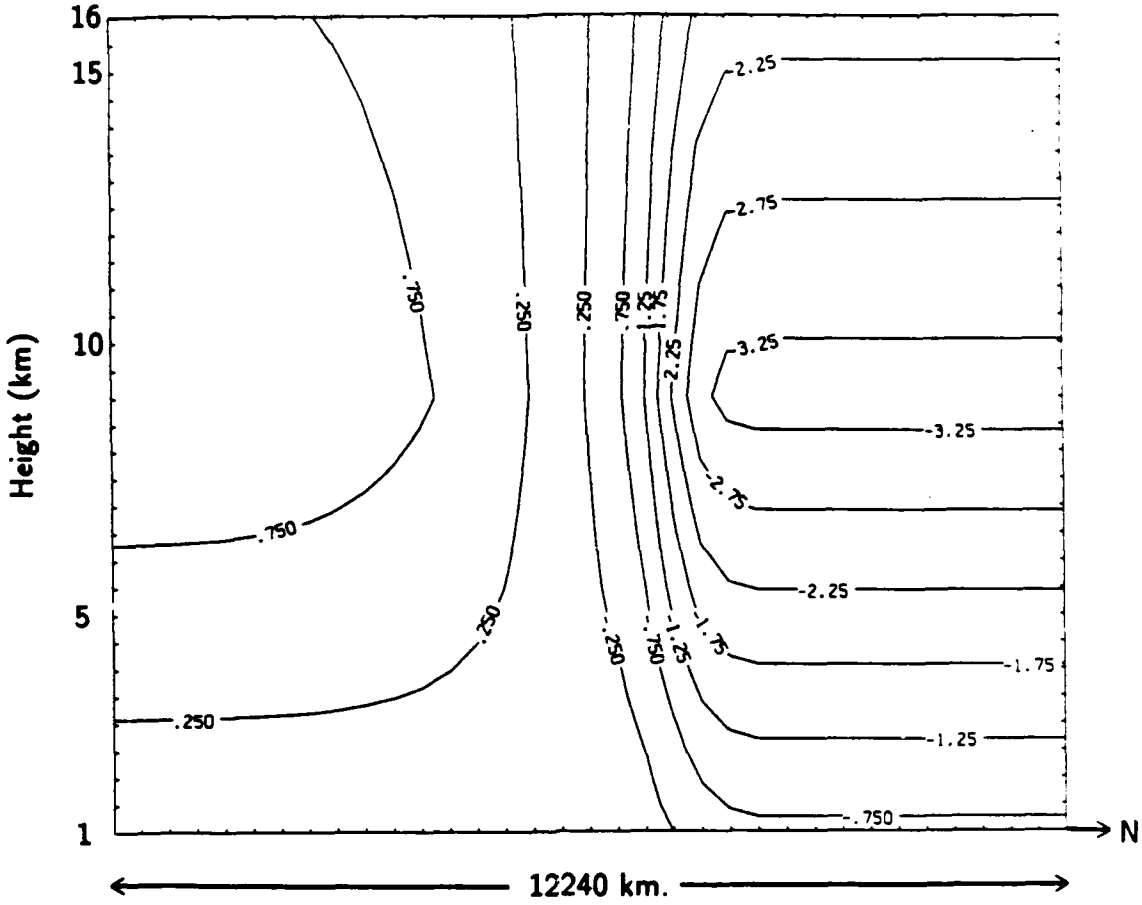


Figure 30B Perturbation pressure ϕ' for the geostrophically balanced initial jet in Figure 30A.

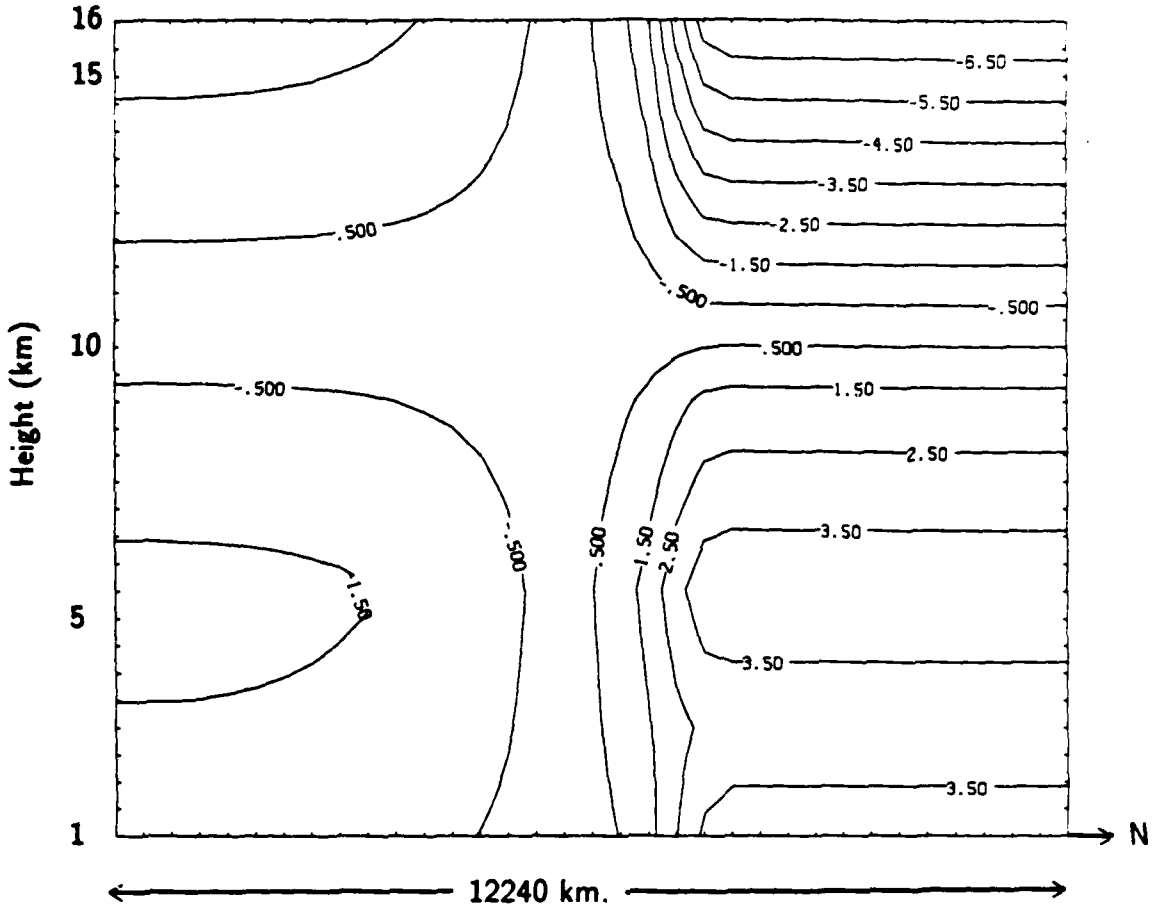


Figure 30C s perturbation (s') for the geostrophically balanced jet in Figure 30A.

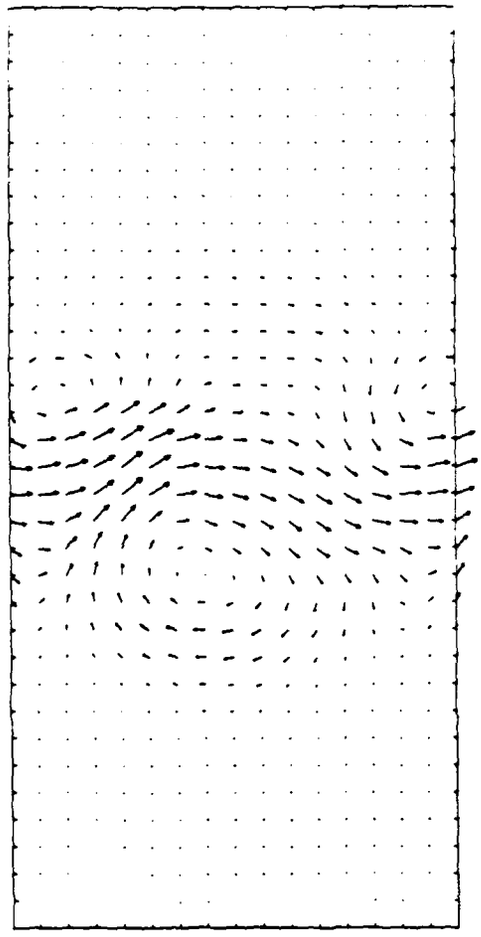


Figure 31 Velocity vectors on the $z = 1$km surface at the start of the adaptive run. Note the circulation about the low and high pressures in Figure 32 and the position of the cold and warm fronts (Figure 33) relative to the vectors.

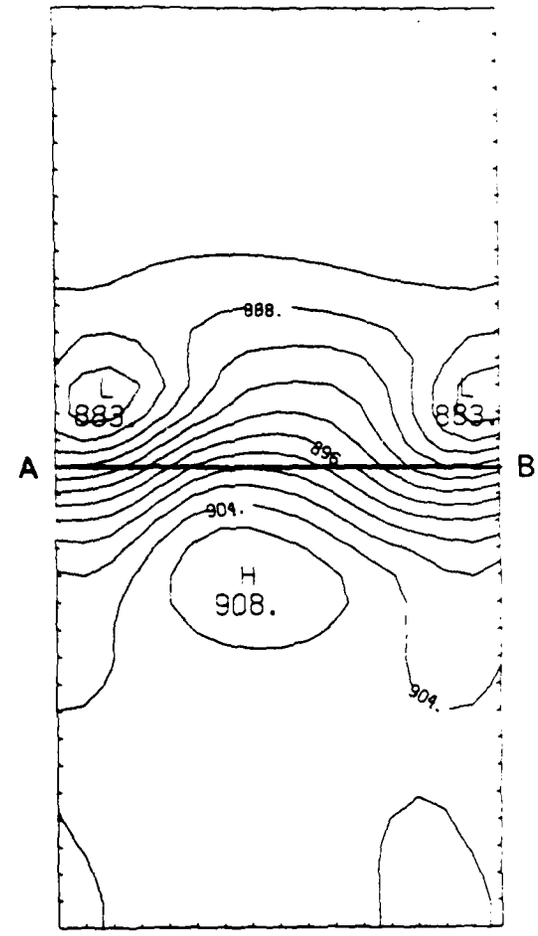


Figure 32 Pressure (millibars) at $z = 1$km at the start of the adaptive run.

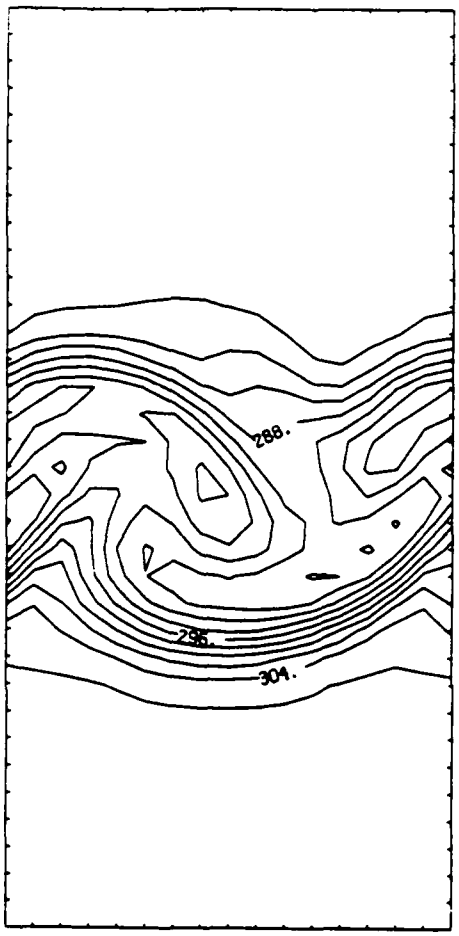


Figure 33 Potential temperature (K) at $z = 1$km at the start of the adaptive run.

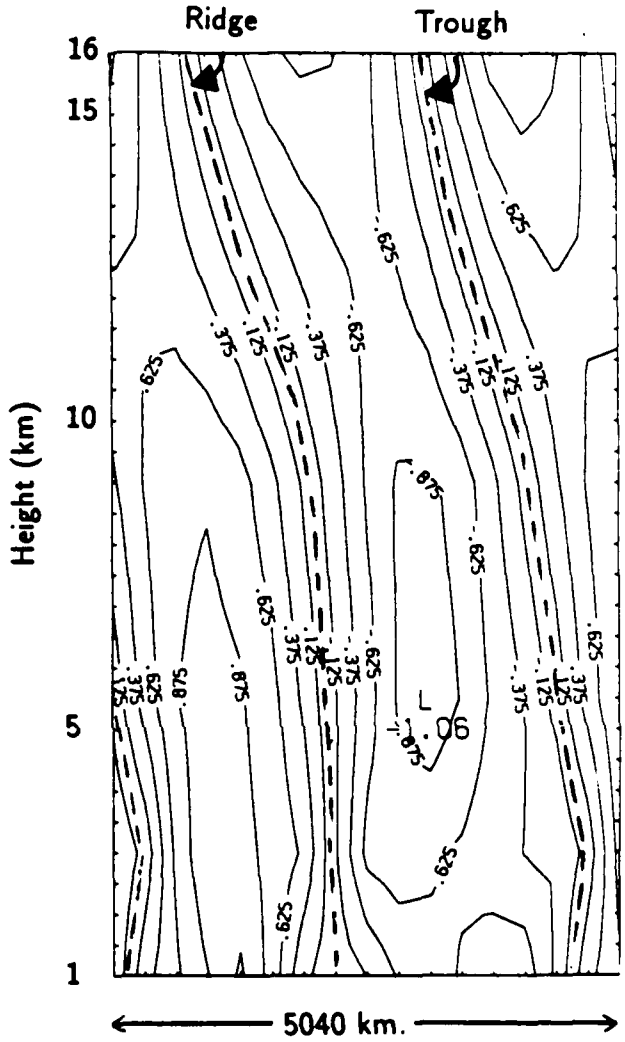


Figure 34 Vertical cross section of the dimensionless velocity v' through AB in Figure 32. Note the westward slope of the ridges and troughes.

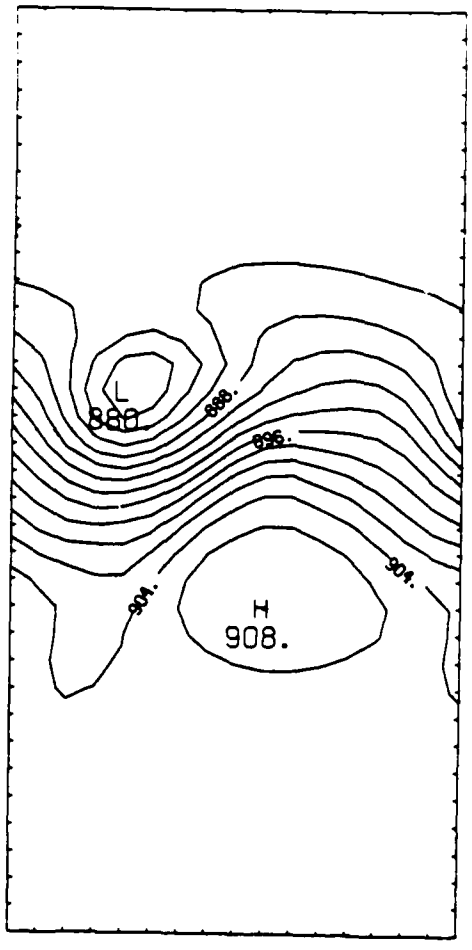


Figure 35 Pressure (millibars) at z = 1km after 36 hours integration time.

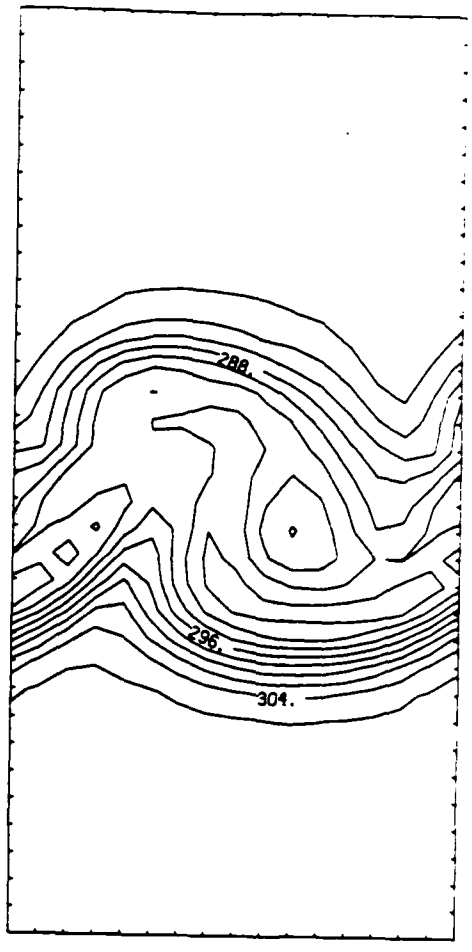


Figure 36 Potential temperature (K) at z = 1km after 36 hours integration time.

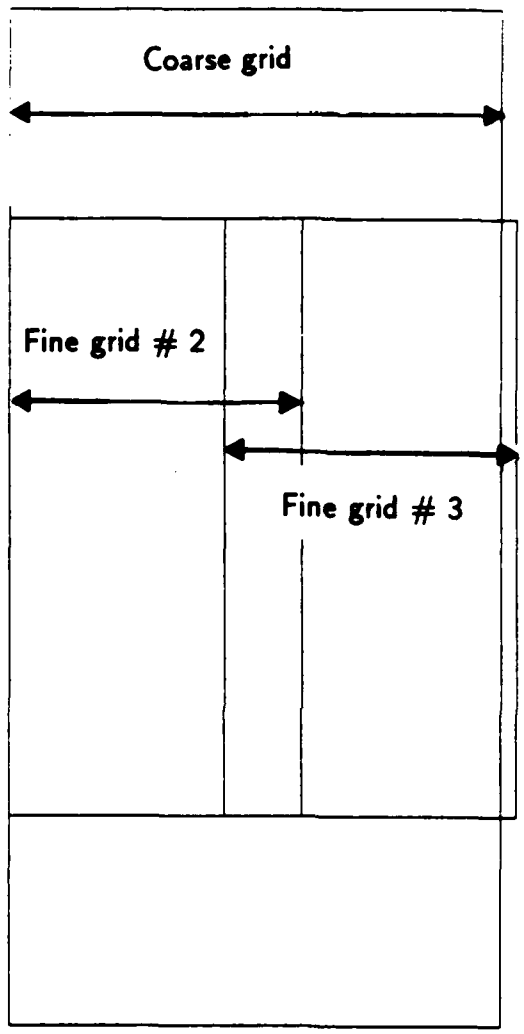


Figure 37 Fine grid positions.

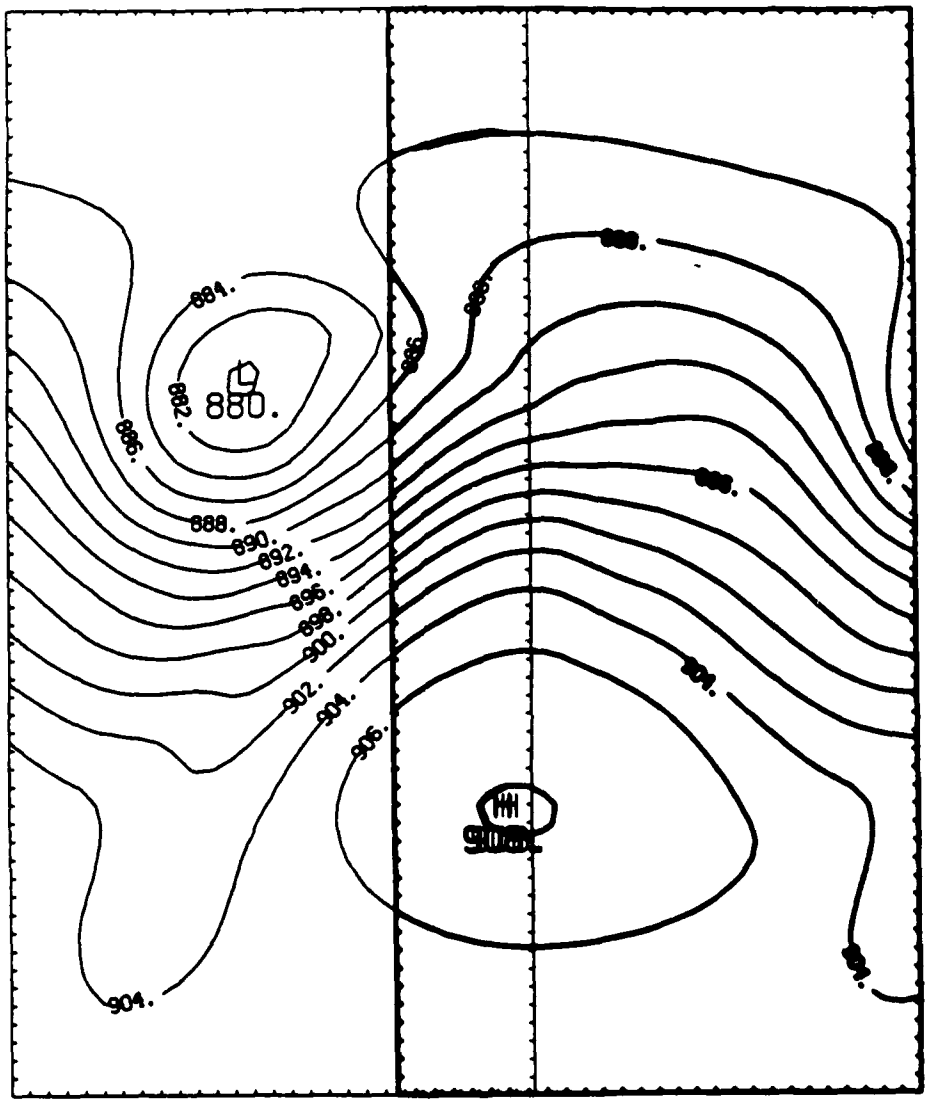


Figure 38 Pressure (millibars) on fine grids 2 and 3 at $z = 1$km. The solutions agree in the overlap regions and no kinks occur along the boundaries.

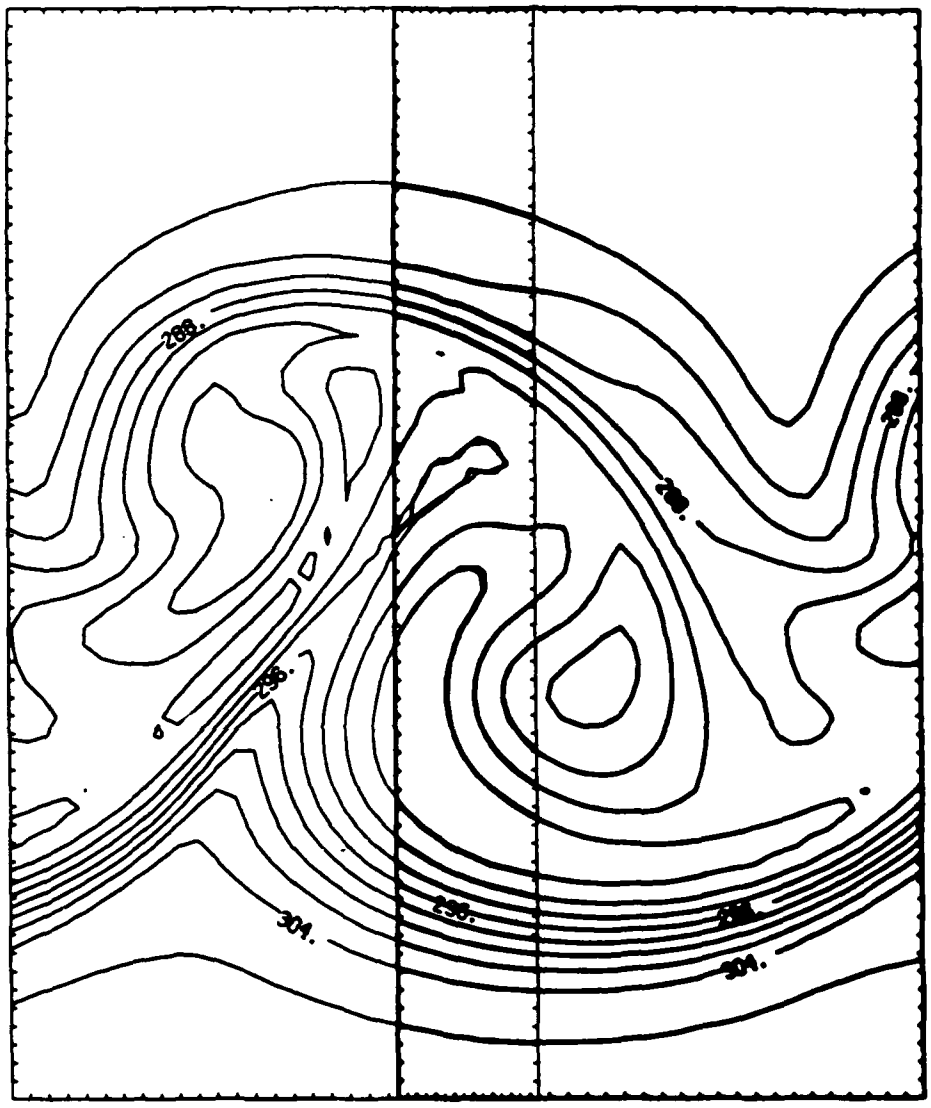


Figure 39 Potential temperature (K) on fine grids 2 and 3 at $z = 1$km. Note the agreement in the overlap region.

7.2 Error Estimation with the Browning-Kreiss Equations

Error estimates using the Richardson procedure (eqn. 2.2) are performed on the fields shown in Figures 31 through 34. As noted earlier, the maximum errors in the velocity fields are found at the jet. We find that the maximum error is associated with the computation of the pressure gradient term in the momentum equation when the BK equations and discretization described in the previous chapter are used. This is in contrast to the results using the primitive equations solver described in Chapter 3. In those results the error was of equivalent size for all the terms.

We can compute the truncation error for the terms in the BK momentum equation 6.5. The terms are already dimensionless and O(1), thus, we can directly compare the error terms. The computed truncation errors are

$$\begin{aligned}
\tau\left(u\frac{\partial u}{\partial x}\right) &= \frac{\Delta x^2}{2}\left(\frac{1}{3}uu_{xxx}\right) + O(\Delta x^3) \\
\tau\left(v\frac{\partial u}{\partial y}\right) &= \frac{\Delta x^2}{2}\left(\frac{1}{4}u_y(v_{xx}+v_{yy}) + \frac{1}{3}vu_{yyy}\right) + O(\Delta x^3) \\
\tau\left(\epsilon^{-n}\frac{\partial \phi}{\partial x}\right) &= \frac{\Delta x^2}{2}\left(\frac{5}{6}\phi_{xxx}\right) + O(\Delta x^3) \\
\tau\left(-\epsilon^{-n}fv\right) &= -\frac{\Delta x^2}{2}\left(\frac{5}{2}f(v_{xx}+v_{yy})\right) + O(\Delta x^3)
\end{aligned}$$

and we have used $\epsilon = 10^{-1}$.

We can compare these truncation errors with those computed for primitive equations (4.1a - 4.1d). The relative size of the errors is similar for the corresponding terms but the forms are not necessarily the same because the forms and discretization of the terms differ.

Error estimates for the pressure are noisy as they were for the surface pressure when calculating with the primitive equations. Once again this is due to the difficulty in accurately calculating the error in the divergence. Error estimates for the temperature show that the regions of high error to lie in regions of large horizontal temperature advection. These estimates, as those for the primitive equations, also tend to be noisy.

We conclude that error estimates are largely dependent upon the equations and their discretization. Error estimates should be re-analyzed when equations or the discretizations are changed.

7.3 Computing Non-hydrostatic Motions

Non-hydrostatic motions can be computed using the BK equations. Outlined in this section is a possible extension of the present adaptive BK model which would allow the computation of non-hydrostatic motions.

There are three parameters in the non-dimensional BK equations (6.3-6.6, 6.8) which depend upon the scales of motion in the flow. These parameters are n, η and α. By using the dimensional version of 6.3-6.6, 6.8 we can eliminate n and η. The dimensional system is

$$\frac{ds}{dt} + \tilde{s}w = 0 \tag{7.1}$$

$$\frac{d\phi}{dt} + p_o\rho_o^{-1}[\gamma d + \tilde{p}w] = 0 \tag{7.2}$$

$$\frac{du}{dt} + \phi_x - fv = 0 \tag{7.3}$$

$$\frac{dv}{dt} + \phi_y + fu = 0 \tag{7.4}$$

$$\frac{dw}{dt} + \alpha\big(\phi_z + 0.1\tilde{s}\phi + gs\big) = 0 \tag{7.5}$$

where all variables are now the dimensional counterparts of the dimensionless variables defined in Chapter 6.

The dimensionless parameter α remains. It must be chosen such that the dimensional counterpart of the elliptic equation for the pressure (6.9) has smooth solutions. This is accomplished when $\alpha = (D/L)^2$. In an adaptive model, where Δx and Δz are appropriate for the scales of motion found on a particular level grid, we can satisfy this constraint by choosing $\alpha = (\Delta z/\Delta x)^2$, hence, α is determined by the grid level. A model using these equations would be capable of simulating hydrostatic large-scale motions and nonhydrostatic, small-scale motions in a single, adaptive computation.

As a final note, we mention that there is still one major problem to be solved before such a computation would be reasonable. It will be necessary to use vertical refinement in an adaptive model which computes both hydrostatic and non-hydrostatic motions. Non-hydrostatic motions have much shorter vertical length scales than their hydrostatic counterparts, and a vertical discretization sufficient to resolve non-hydrostatic motions would not prove practical for the larger scale computations.

8. CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions

An adaptive grid refinement technique has been used to compute solutions to equations describing large-scale atmospheric flow. Fine grids are placed automatically based on a Richardson-type estimate of the local truncation error in the solution. Simulations of a barotropic cyclone and of a baroclinically unstable jet were performed to demonstrate the feasibility of using techniques of this type in NWP and similar large-scale flow computations.

Successful simulations with both the primitive equations and the Browning-Kreiss equations strongly support the concept that refinement need only occur only where dictated by the error in the numerical solution. This is sufficient to improve the accuracy and overall resolution of the entire solution. Using this simple concept has produced the first adaptive solution of atmospheric flows and the first detailed, quantitative results concerning the truncation error in the numerical solutions. These simulations represent the first adaptive solutions of three-dimensional time-dependent fluid flow.

Several critical components ensure an accurate, smooth solution. Numerical schemes must be consistent up to the boundaries. Changing operators close to the boundaries may cause kinks and discontinuities. When fine grids overlap, boundary values for one fine grid must come from the other. This necessitates the use of a fully explicit scheme (explicit even with respect to the boundary conditions) or of some new scheme which would take into account the overlapping fine grid constraint. Higher order interpolation techniques for use in setting the initial conditions are necessary so as not to excite gravity waves when initializing any fine grid fields.

Richardson estimates of the truncation errors in the u and v momentum equations compare well with the directly computed truncation errors for both the primitive equations and the Browning-Kreiss equations. Error estimates for the pressure for both sets of equations are unreliable and noisy because it is difficult to obtain accurate, smooth values for the local mass divergence and even more difficult to compute the errors in the mass divergence.

Truncation errors arising from the spatial discretization dominate the overall

truncation error. For the primitive equations solver the error was equidistributed among the terms in the equations. In the BK solver the dominant terms in the truncation error arose from the terms describing the geostrophic balance, i.e., the large terms in the equations. The difference in the truncation errors between the two solvers indicates that analyses of the error estimates should be performed on a case by case basis. Even though in general, the errors are large in regions where we would expect them to be large, the form and magnitude of the errors depend upon the equations and discretization used.

Only 2-D horizontal refinement was successfully implemented. Several problems arose when we attempted to introduce uniform vertical refinement into the adaptive method. The most serious problem resulted from the vertical interpolation of the thermodynamic variables. We have yet to discover a method which will return smooth, statically stable profiles, will preserve the hydrostatic and geostrophic balances present in large-scale flows and is suitable for use in an adaptive or nested model.

8.2 Recommendations

There are several directions future research may take in light of the results obtained to date. An adaptive model for large-scale atmospheric flows, a model which includes realistic physics, should be developed and tested with real data. Only in this way, by testing with actual data and with analyses of the model's predictive capabilities, will the true worth of the adaptive method become known.

A more immediate research problem connected with developing an operational, adaptive, large-scale atmospheric flow solver is the development of a splitting scheme for use in an adaptive model. In our simulations we used a fully explicit solver because fully explicit boundary conditions were needed in the case where fine grids overlap and so that continuity boundary conditions could be successfully applied at fine-coarse grid boundaries. Splitting methods can allow the use of time steps 5 to 10 times larger than that used in an explicit technique with little added cost per time step. Even though the overhead intrinsic to the adaptive method is relatively small, the development of a splitting technique for adaptive use may well prove critical in the development of a large-scale operational adaptive atmospheric model.

An adaptive model should be developed for the computation of non-hydrostatic flows. We have briefly outlined how this might be accomplished with the Browning-Kreiss adaptive model. Other equation sets used for non-hydrostatic motions might also be tried. Some of the more important questions remaining in atmospheric science exist in the study of smaller scale motions (mesoscale meteorology). The next generation of mesoscale models will play a large, if not the largest, role in answering these questions and the models will certainly need to incorporate some adaptive or greatly expanded nested capability.

The vertical refinement problem needs further study. The use of adaptive or nested models in the study of several interacting scales of motion will be limited unless an answer to this problem is found, especially if the scales span both hydrostatic and non-hydrostatic motions.

APPENDIX 1

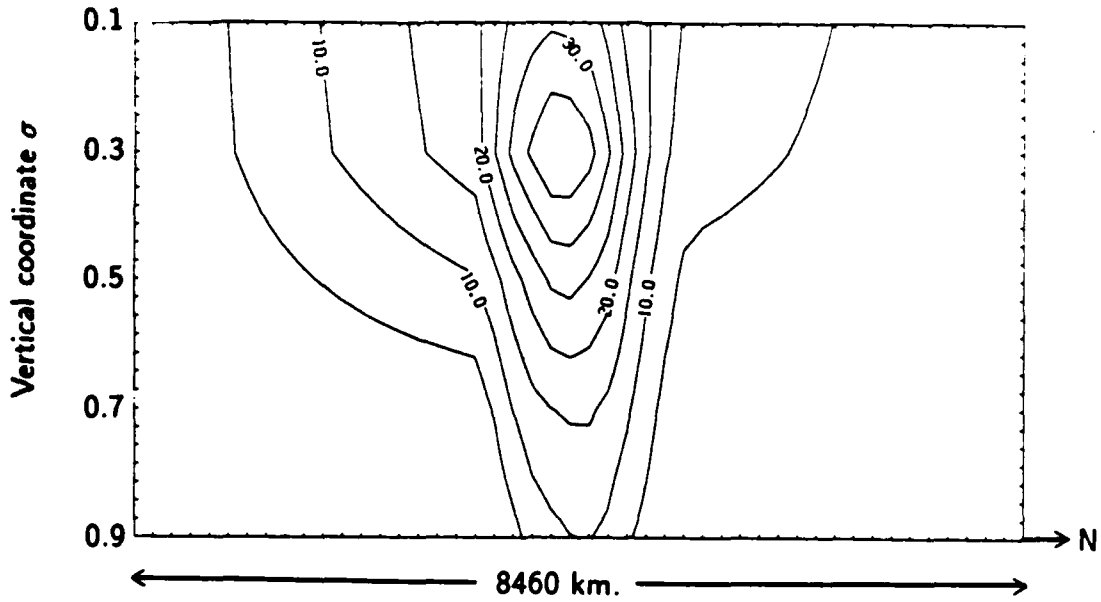
Additional Results

In this appendix we present additional plots of the results obtained with the adaptive primitive equations solver. The results are for the case described in Chapter 4, Section 2 unless otherwise noted.

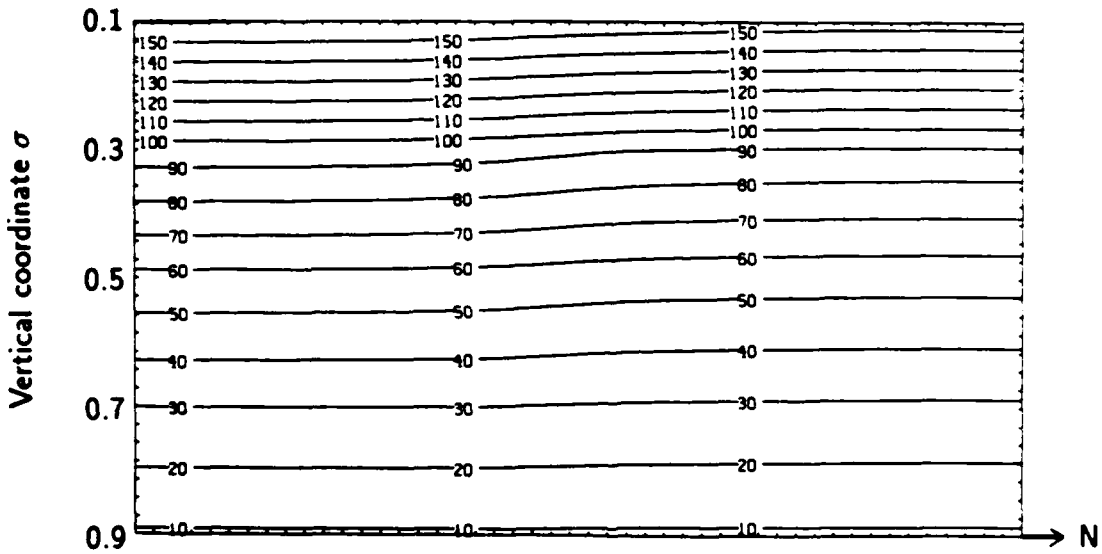
The first figures shown are plots of the u velocity, temperature, geopotential and surface pressure for the baroclinically unstable jet before it is perturbed. The jet is geostrophically balanced and the atmosphere is statically stable.

The next set of figures shows the geopotential at 500 mb. and the magnitude of the horizontal velocity on the 300 mb. surface (where the jet core is located) after 72 hours of the adaptive test run. Other fields are plotted in Chapter 4. The 500 mb. field clearly shows the developing wave and further illustrates the poor resolution of the coarse grid solution compared to the solutions on the fine and adaptive grids. The maximum velocities at the jet core are twice as large in the fine and adaptive runs compared with the coarse grid run.

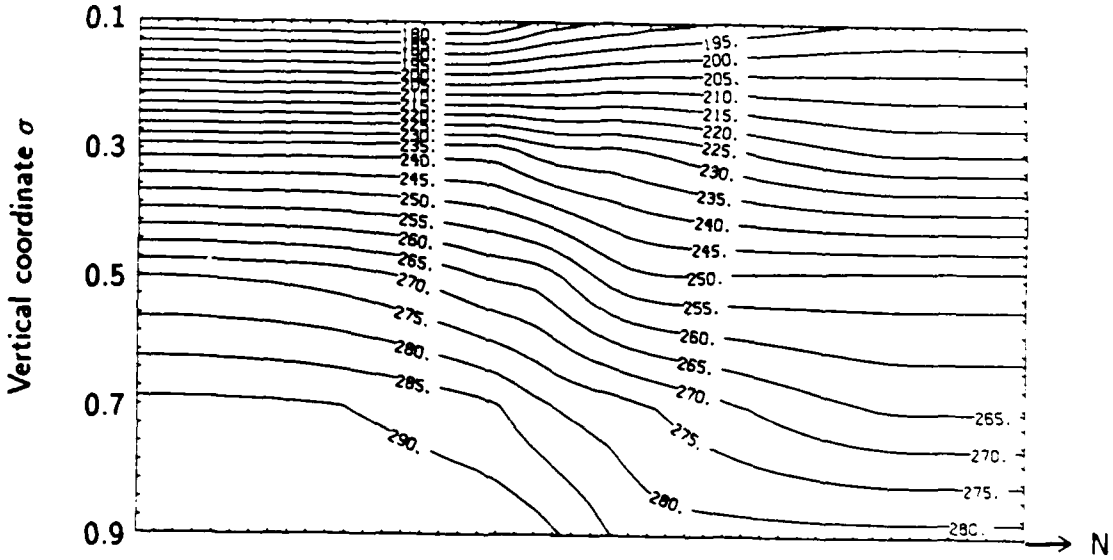
The test runs were carried out to 6 days. Plots of the surface pressure, vorticity and geopotential at 500 mb. after 6 days illustrate the continuing decay of the coarse grid run solution. The surface lows are still gaining strength in the fine and adaptive grid runs though the disturbance is in its mature phase.



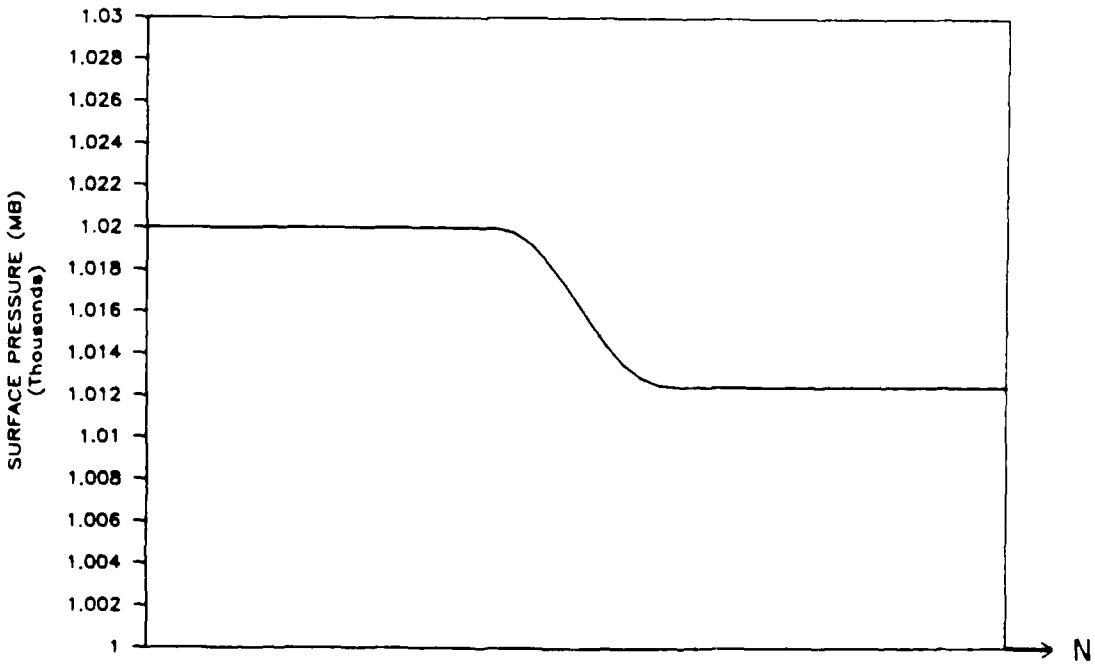
Initial jet u velocity (m/s), S-N cross section.



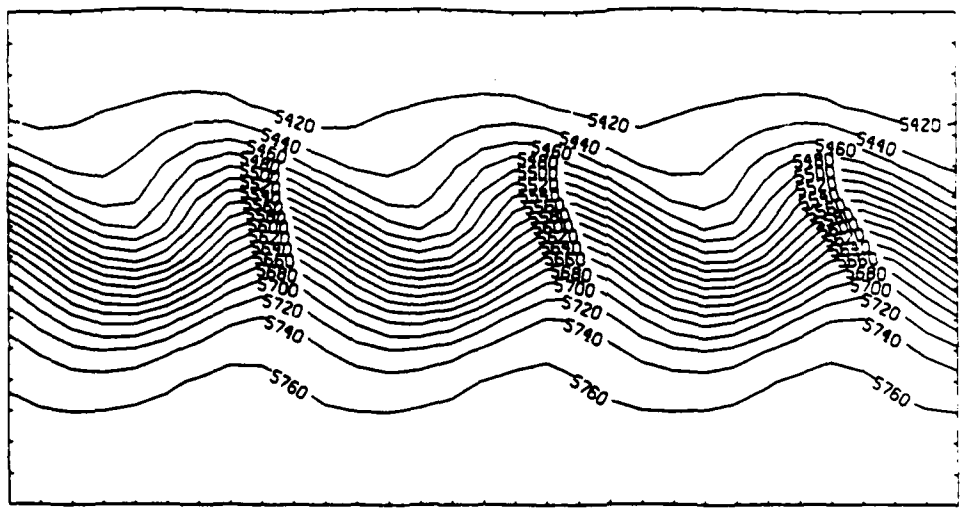
Geopotential ($m^2/s^2 \times 10^{-4}$) for the geostrophically balanced initial jet.



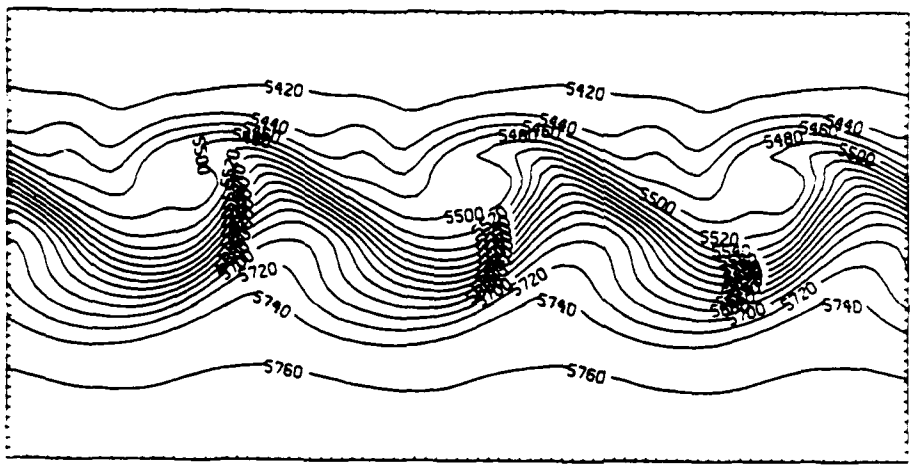
Temperature (K) for the geostrophically balanced initial jet.



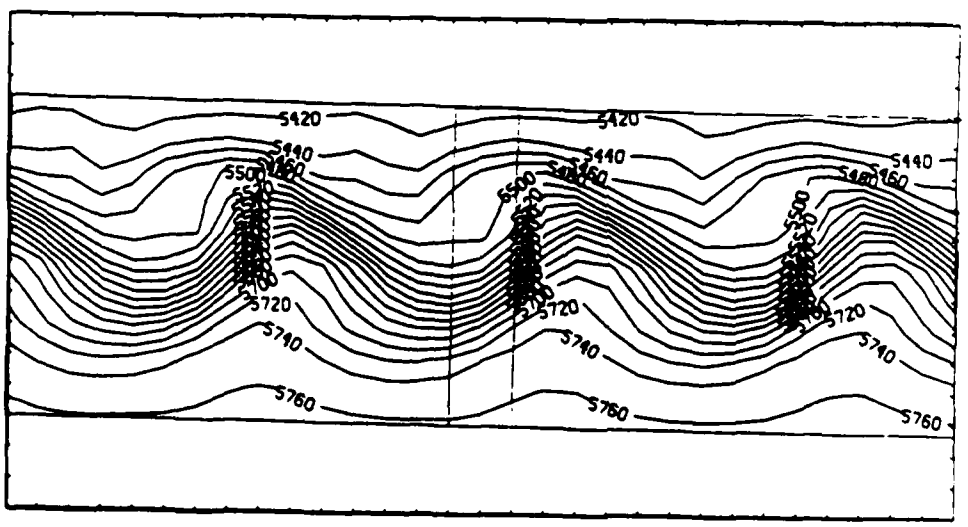
Surface pressure (mb) the geostrophically balanced initial jet.



COARSE GRID RUN

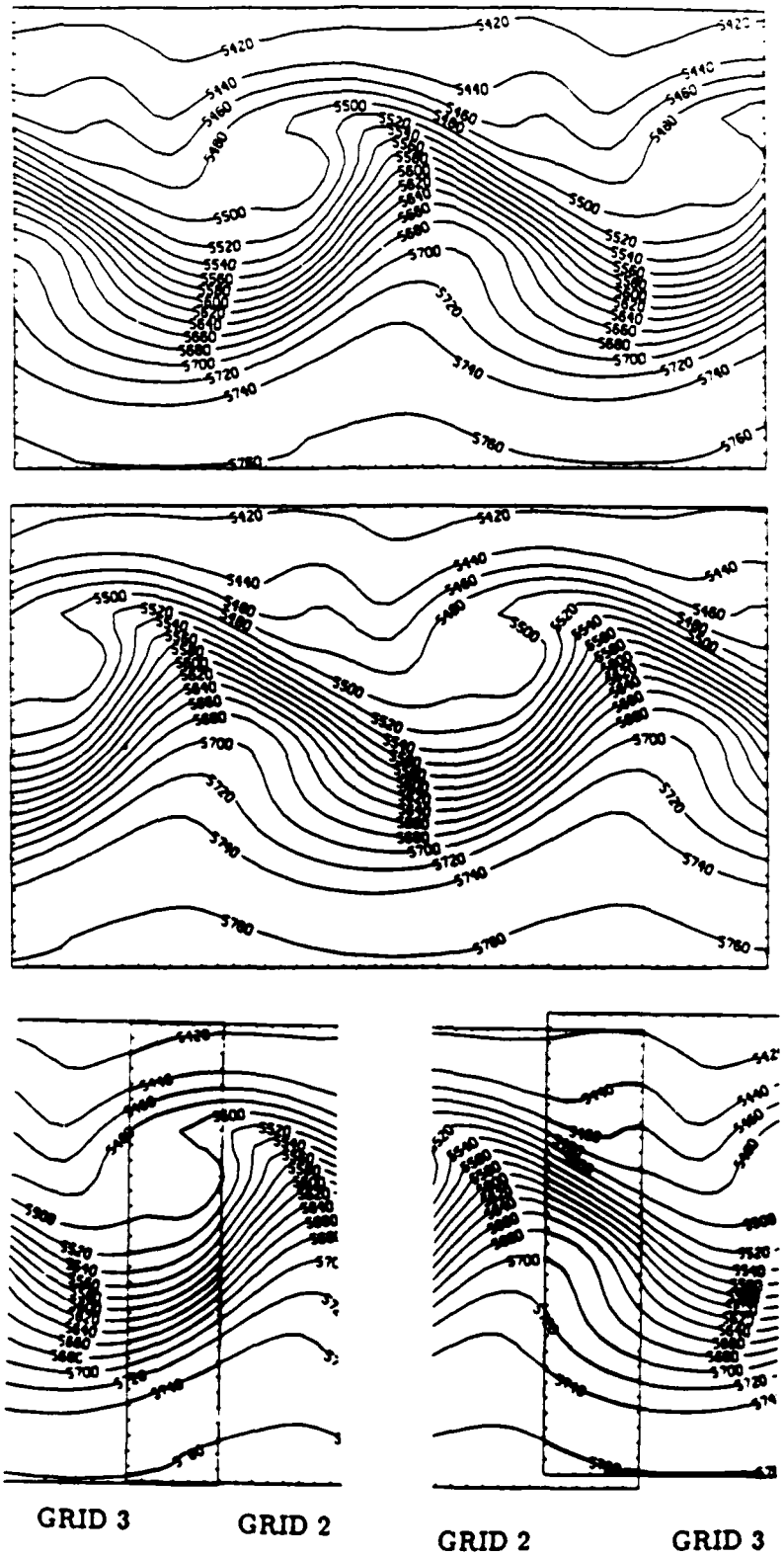


FINE GRID RUN

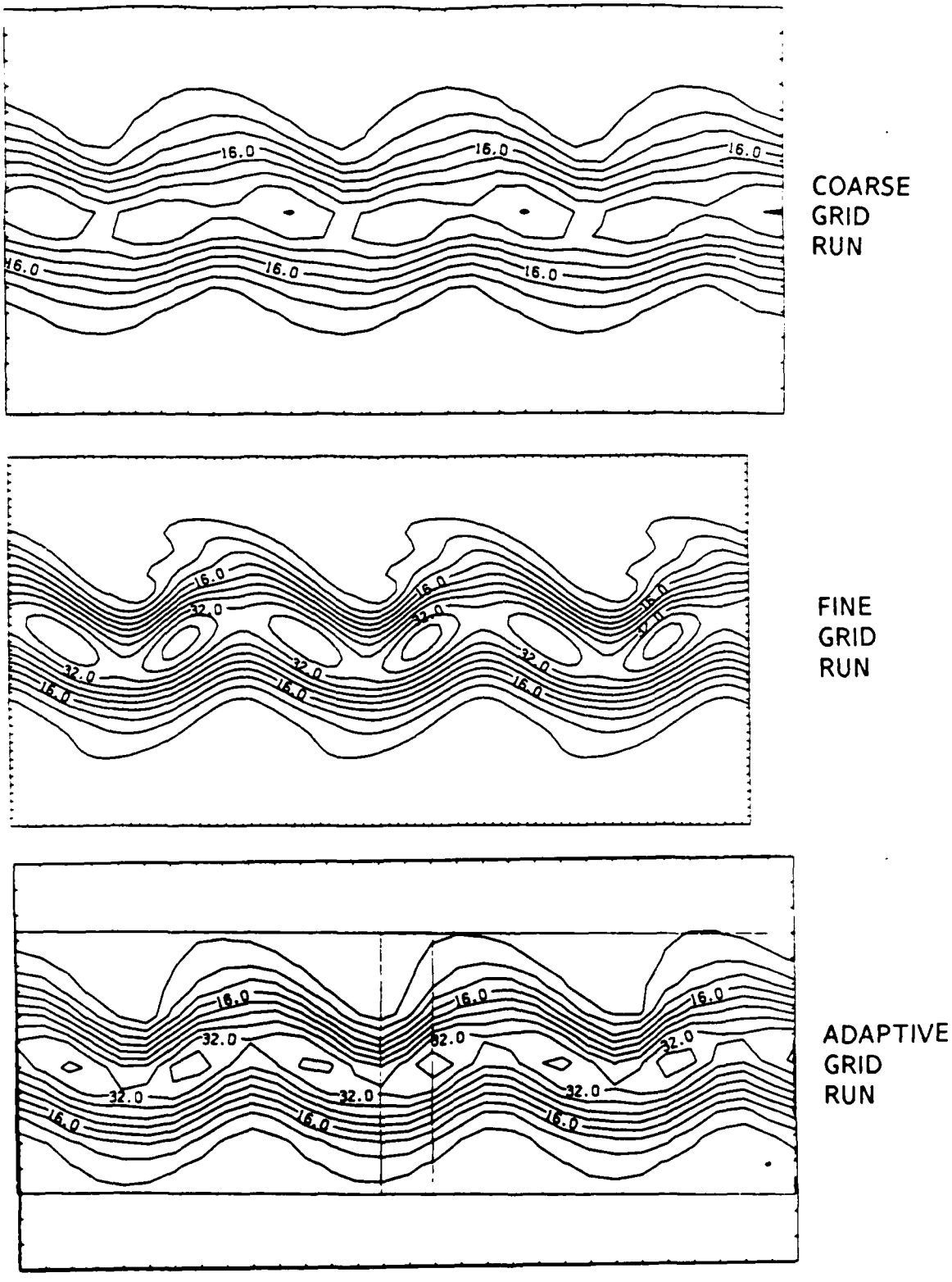


ADAPTIVE GRID RUN

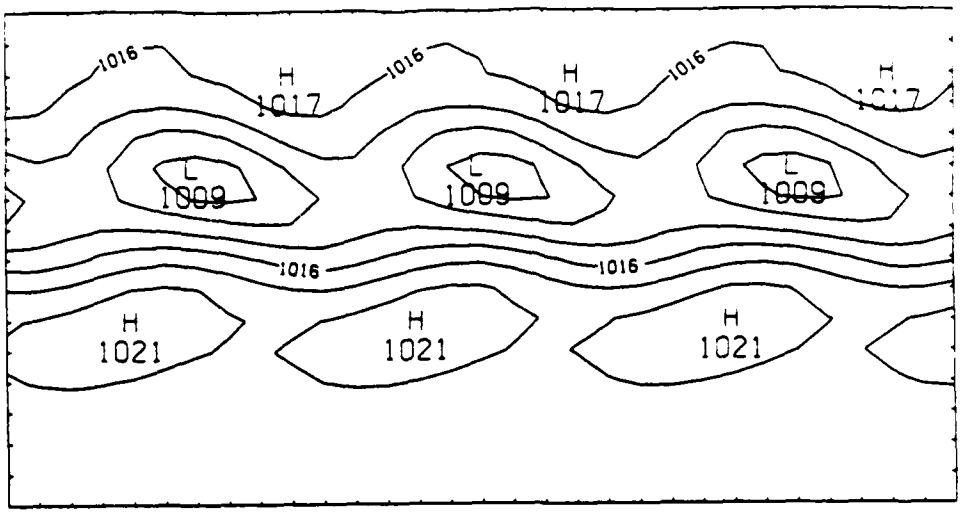
Height (meters) of the $\sigma = 0.5$ surface at $T = 72$ hours.



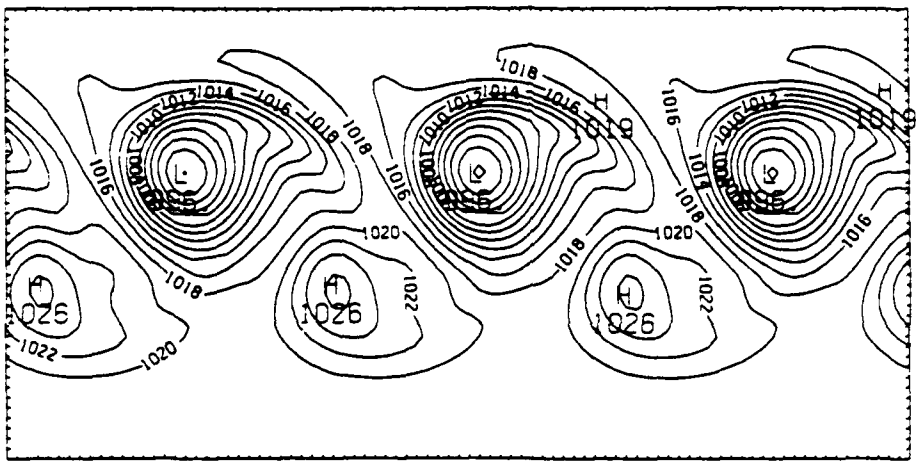
Height (meters) of the $\sigma = 0.5$ surface at $T = 72$ hours for the adaptive fine grids 3 (top), 2 (middle) and their overlaps (bottom).



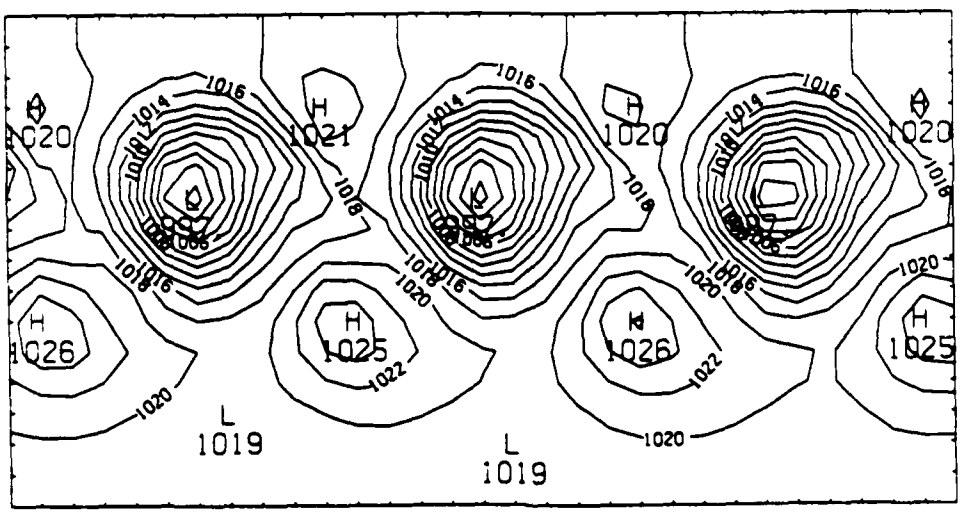
Magnitude of the horizontal velocity ($||V_h||$, m/s) on the $\sigma = 0.3$ surface at $T = 72$ hours.



COARSE GRID RUN

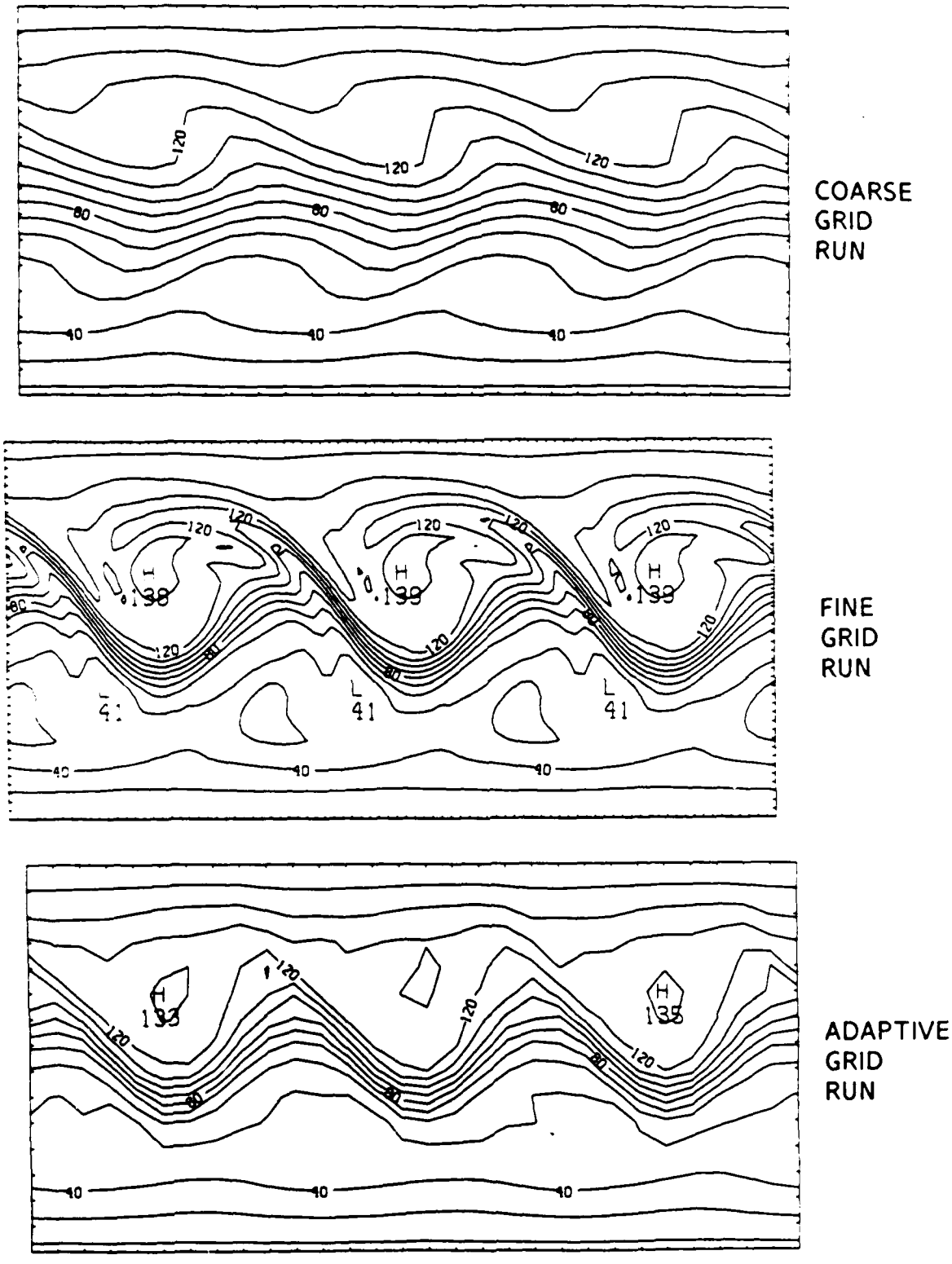


FINE GRID RUN

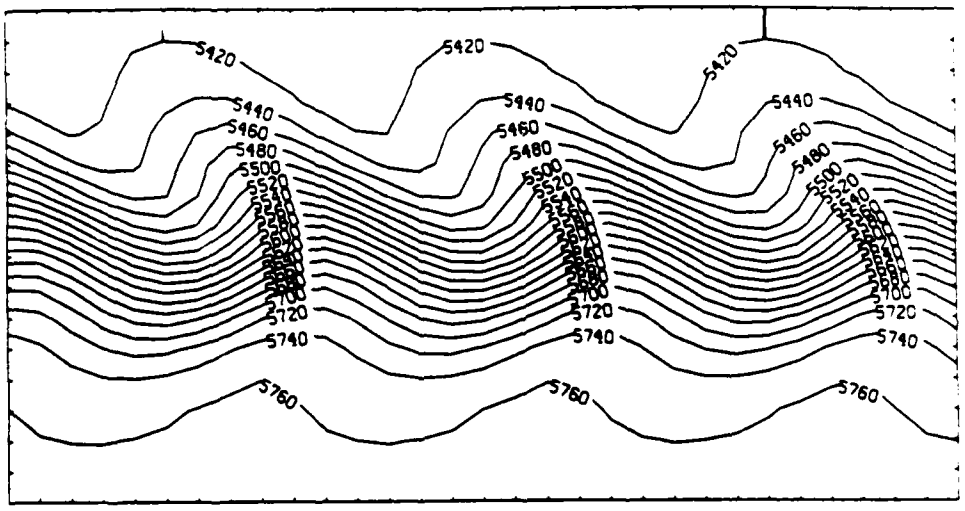


ADAPTIVE GRID RUN

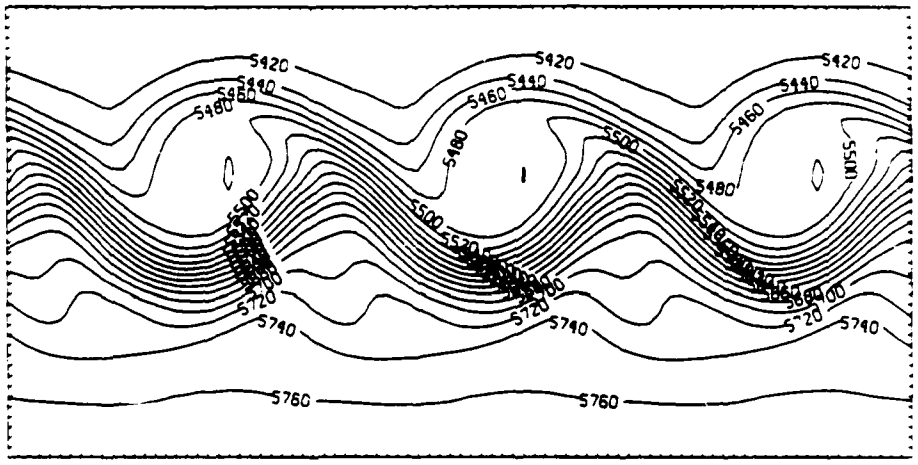
Surface pressure (mb) at $T = 144$ hours.



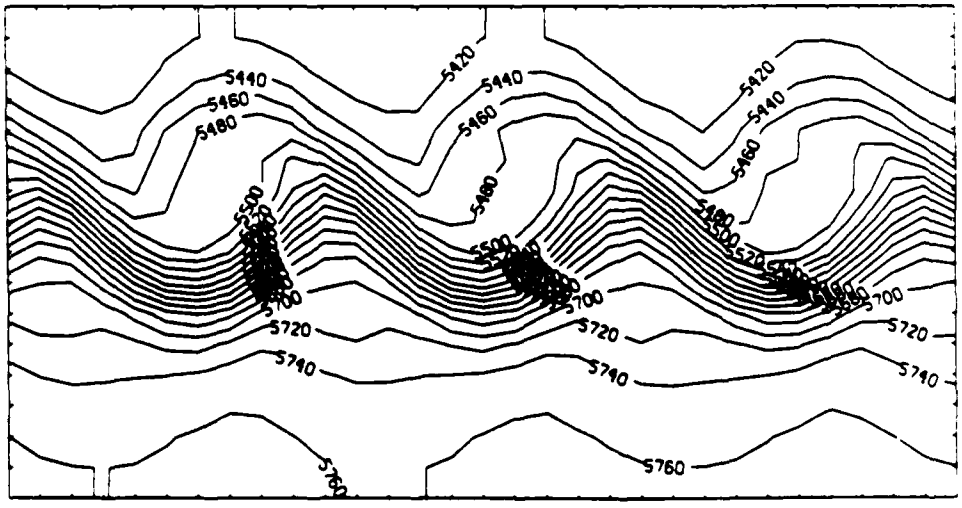
Absolute vorticity ($\times 10^{-6} s^{-1}$) on the $\sigma = 0.5$ surface at $T = 144$ hours.



COARSE GRID RUN



FINE GRID RUN



ADAPTIVE GRID RUN

Height (meters) of the $\sigma = 0.5$ surface at $T = 144$ hours.

APPENDIX 2

Truncation Error for the PE Momentum Equations

We wish to estimate the relative size of the truncation errors associated with the spatial discretization. This can be accomplished by scaling and nondimensionalizing the equations and the truncation error associated with the discretization. In this model we do not refine in the vertical and the Richardson error estimate does not estimate errors in the vertical differencing. Thus, to simplify the analysis we do not consider the vertical advection terms.

For our purposes it is sufficient to analyze only one momentum equation. We scale and nondimensionalize the u momentum equation (3.1) with the following change of variables:

$$
\begin{aligned}
\pi &= \pi_o + \hat{\pi}\pi' & \pi_o &= 1000\text{mb.} \\
& & \hat{\pi} &= 10\text{mb.} \\
\phi &= \phi_o + \hat{\phi}\phi' & \phi_o &= 10^5\text{m}^2/\text{s}^2 \\
& & \hat{\phi} &= 10^3\text{m}^2/\text{s}^2 \\
T &= T_o + \hat{T}T' & RT_o &= 10^5\text{m}^2/\text{s}^2 \\
& & R\hat{T} &= 10^4\text{m}^2/\text{s}^2 \\
u &= Uu' & U &= 10\text{m/s} \\
v &= Uv' & & \\
x &= Lx' & L &= 10^6\text{meters} \\
y &= Ly' & & \\
t &= t_o t' & t_o &= L/U = 10^5\text{s} \\
f &= f_o f' & f_o &= 10^{-4}\text{s}^{-1}
\end{aligned}
$$

where the primed variables are dimensionless and of O(1). The scaling values are appropriate for large-scale atmospheric flows. If we substitute these into equation 3.1, drop the obviously lower order terms and divide trhough by $\pi_o U/t_o$ we arrive at the following nondimensional momentum equation:

$$
\frac{\partial u'}{\partial t'} = -\underbrace{\frac{Ut_o}{L}}_{O(1)}\left(\frac{\partial(u'u')}{\partial x'} + \frac{\partial(u'v')}{\partial y'}\right) - \underbrace{\frac{t_o\hat{\phi}}{UL}}_{O(10)}\frac{\partial \phi'}{\partial x'} - \underbrace{\frac{\hat{\pi}t_o}{\pi_o UL}RT_o}_{O(10)}\frac{\partial \pi'}{\partial x'} + \underbrace{f_o t_o}_{O(10)} f'v'
$$

Using these scalings we find that, for large-scale atmospheric flows, the Coriolis term

must balance the pressure gradient terms and the advection terms are an order of magnitude smaller than either of these. This simply describes the geostrophic nature of the flow.

We can now use these results to scale and nondimensionalize the truncation error associated with the spatial discretization of equation 3.1. The leading order terms ($O(\Delta x^2)$) of the dimensional truncation error associated with the terms in equation 3.1 are:

$$
\begin{aligned}
\tau\big(\partial(\pi uu/)\partial x\big) =& \frac{\Delta x^2}{4}\big(\frac{4}{3}\pi u u_{xxx} + \frac{7}{6}uu\pi_{xxx} + 3u\pi_x u_{xx} + 4uu_x\pi_{xx} \\
&+ 2\pi u_x u_{xx} + 2\pi_x u_x^2\big) + O(\Delta x^3) \\
\tau\big(\partial(\pi uv)/\partial y\big) =& \frac{\Delta x^2}{4}\big(uv_{xy}\pi_x + \pi_x v_x u_y + u\pi_{xy}v_x + \frac{1}{6}\pi u v_{yyy} + \frac{1}{2}\pi u_y v_{yy} \\
&+ \frac{1}{2}u\pi_y v_{yy} + \pi u_{yy}v_y + \pi_y u_y v_y + \frac{1}{2}u\pi_{xx}v_y + u\pi_{yy}v_y \\
&+ \frac{2}{3}\pi v u_{yyy} + v\pi_y u_{yy} + \frac{1}{2}v\pi_{xx}u_y + v\pi_{xx}u_y + \frac{1}{2}uv\pi_{xxy} \\
&+ \frac{2}{3}uv\pi_{yyy} + \pi u v_{xxy} + \pi u_y v_{xx}\big) + O(\Delta x^3) \\
\tau\big(\partial(\pi\phi)/\partial x\big) =& \frac{\Delta x^2}{4}\big(\frac{1}{6}\phi\pi_{xxx} + \frac{1}{6}\pi\phi_{xxx} + \frac{1}{2}\phi_x\pi_{xx} + \frac{1}{2}\phi_{xx}\pi_x\big) + O(\Delta x^3) \\
\tau\big((RT-\phi)\partial\pi/\partial x\big) =& \frac{\Delta x^2}{4}\big(\frac{1}{6}(RT-\phi)\pi_{xxx} + \pi_x(RT_{xx} - \phi_{xx})\big) + O(\Delta x^3) \\
\tau(f\pi v) =& \frac{\Delta x^2}{4}f\big(\frac{1}{2}v\pi_{xx} + \frac{1}{2}\pi(v_{xx} + v_{yy}) + \pi_x v_x\big) + O(\Delta x^3)
\end{aligned}
$$

We nondimensionalize the truncation errors by substituting the previously defined primed variables and dividing by $U\pi_o/t_o$. The leading order nondimensional terms are given in Chapter 4, Section 4.3. The nondimensionalization of the truncation errors indicate that all are of the same size, even though the respective terms from which the truncation errors are derived are not.

APPENDIX 3

Discretization of the Browning-Kreiss Equations

In this appendix we outline the discretization of the Browning-Kreiss equations (6.3 – 6.6, 6.8). A time-centered, space-centered discretization is used. The time marching scheme is the leapfrog method described in Chapter 3. We outline the spatial discretization here.

Interior Differencing

The spatial discretization is performed on the C-grid shown in Figure 6. First we examine the advection terms. For ϕ and s the horizontal advection terms are differenced as

$$u\frac{\partial(\phi,s)}{\partial x}\bigg|_x \approx \frac{1}{2}(u_{i-\frac{1}{2}}+u_{i+\frac{1}{2}})\frac{(\phi,s)_{i+1}-(\phi,s)_{i-1}}{2\Delta x}$$

where $x = i\cdot\Delta x$. The differencing is similar, only rotated, for the $(\phi,s)_y$ terms.

For the horizontal velocity advection terms the differencing is straightforward. We difference the advection terms as follows.

$$u\frac{\partial u}{\partial x}\Big|_x \approx u_i\frac{u_{i+1}-u_{i-1}}{2\Delta x}$$

This differencing can be rotated for the vv_y term. The two remaining terms require interpolating the advecting velocity. The differencing is

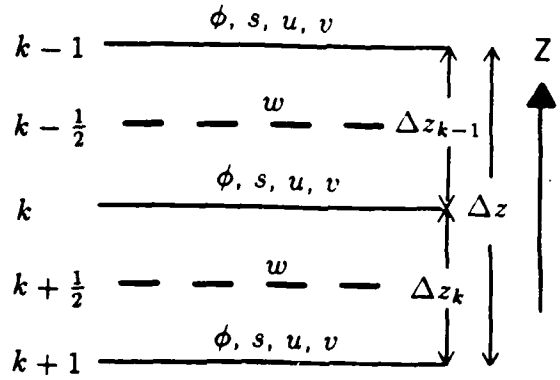
$$v\frac{\partial u}{\partial y}\bigg|_{x,y} \approx \frac{1}{4}(v_{i-\frac{1}{2},j+\frac{1}{2}}+v_{i-\frac{1}{2},j-\frac{1}{2}}+v_{i+\frac{1}{2},j+\frac{1}{2}}+v_{i+\frac{1}{2},j-\frac{1}{2}})\frac{(u_{j+1}-u_{j-1})}{2\Delta y}$$

where $x = i\cdot\Delta x$ and $y = j\cdot\Delta y$. Again, the differencing for the uv_x term can be obtained by rotation.

The differencing of the vertical advection terms is complicated by the fact that the grid is not uniformly staggered in the vertical. The vertical velocities are located along the same vertical axis as ϕ and s. We compute the vertical advection term by first interpolating the advecting velocity and then by computing the vertical derivative.

$$w_k = \frac{1}{2}\left(\frac{\Delta z_{k-1}}{\Delta z} w_{k-\frac{1}{2}} + \frac{\Delta z_k}{\Delta z} w_{k+\frac{1}{2}}\right)$$

$$w \frac{\partial(\phi, s)}{\partial z}\bigg|_k \approx w_k \frac{(\phi, s)_{k-1} - (\phi, s)_{k+1}}{\Delta z}$$



We must interpolate in both the horizontal and the vertical when computing the advecting velocity in the vertical advection term for the horizontal velocities. w does not lie directly above or below u or v. For this case we can compute the advecting velocity w_k for the horizontal velocity u by using the formula for w_k above and replacing

$$w_{k-\frac{1}{2}} \rightarrow \tfrac{1}{2}\left(w_{k-\frac{1}{2},i-\frac{1}{2}} + w_{k-\frac{1}{2},i+\frac{1}{2}}\right)$$
$$w_{k+\frac{1}{2}} \rightarrow \tfrac{1}{2}\left(w_{k+\frac{1}{2},i-\frac{1}{2}} + w_{k+\frac{1}{2},i+\frac{1}{2}}\right)$$

and using these values for computing the vertical advection of u in the previous formula.

The Coriolis parameter f is located at the ϕ-s points. The differencing for the Coriolis term $-fv|_{i,j}$, where point (i,j) is a u velocity point, is

$$-fv|_{i,j} \approx -\frac{1}{4}\left[f_{i-\frac{1}{2}}\left(v_{i-\frac{1}{2},j+\frac{1}{2}} + v_{i-\frac{1}{2},j-\frac{1}{2}}\right) + f_{i+\frac{1}{2}}\left(v_{i+\frac{1}{2},j+\frac{1}{2}} + v_{i+\frac{1}{2},j-\frac{1}{2}}\right)\right]$$

Rotation and a sign change results in the discretization for $+fu$.

The parameters $\tilde{p}(z)$, $\tilde{s}(z)$ and $p_o(z)/p_o(z)$ are carried at the C-grid levels , i.e., the levels where u, v, ϕ and s are defined. Consequently, since w levels lie halfway between C-grid levels, we difference the hydrostatic terms in the vertical momentum equation in the following manner.

$$(\phi_z + 0.1\tilde{s}\phi + gs)\big|_{z_{k+\frac{1}{2}}} \approx \frac{\phi_k - \phi_{k+1}}{\Delta z_k} + \frac{1}{4}(\tilde{s}_k + \tilde{s}_{k+1}) \cdot (\tilde{\phi}_k + \tilde{\phi}_{k+1}) + \frac{g}{2}(s_k + s_{k+1})$$

Other terms in the equations (6.3–6.6,6.8) are differenced in a manner similar and consistent with what has been presented.

Boundary Conditions

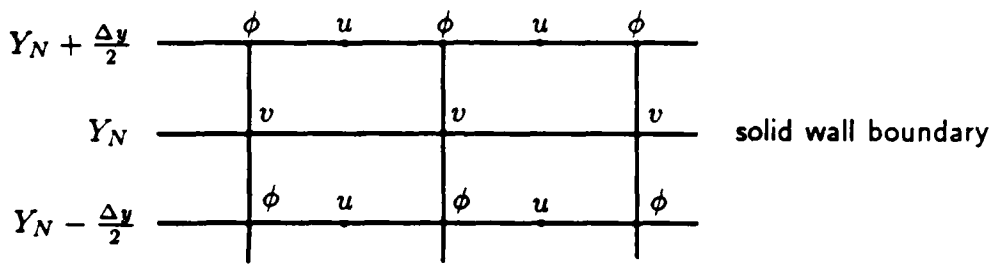
Two types of boundary conditions are applied at the horizontal boundaries of the channel. The East and West boundaries are periodic. We overlap the grid by $2\Delta x$ at this boundary. The North and South boundaries are solid walls and at these walls we require

$$v|_{y=Y_S,Y_N} = 0$$

and

$$\left.\frac{\partial}{\partial y}\right|_{y=Y_S,Y_N} (u, w, \phi, s) = 0.$$

These conditions are discretized by defining fictitious points outside the boundary. For a solid wall at $y = Y_N$



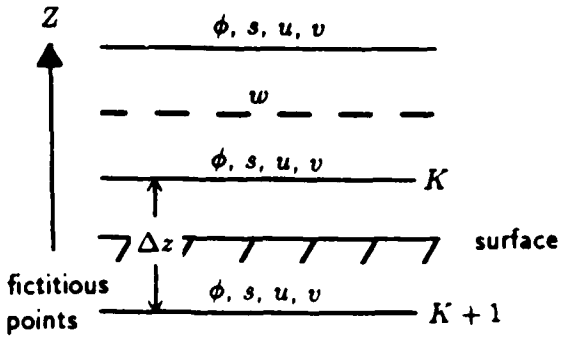
we require that

$$(u, w, \phi, s)|_{Y_N+\Delta y/2} = (u, w, \phi, s)|_{Y_N-\Delta y/2}$$

and that

$$v|_{Y_N} = 0.$$

At the surface we need to specify conditions other than $w = 0$ in order to compute the vertical advection terms at the points closest to the bottom boundary. The bottom boundary is located at $z = 0$. Here we require that the derivatives with respect to z of u, v and s are zero, and, as with the discretization of the lateral solid wall boundaries, we define a fictitious point outside the domain.



Since $(u,v,s)_z = 0$ at $z = 0$, $(u,v,s)_{K+1} = (u,v,s)_K$. ϕ is computed by noting that the atmosphere is hydrostatic at $z = 0$. The hydrostatic equation for the system is

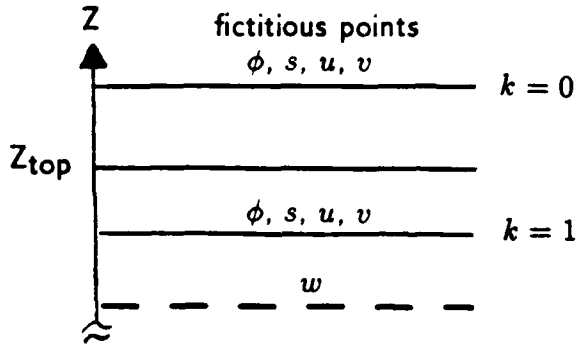
$$(\frac{\partial \phi}{\partial z} + 0.1\tilde{s} + gs) = 0.$$

Applying the hydrostatic constraint at the lower boundary leads to a formula for ϕ_{k+1},

$$\phi_{k+1} = \left[\phi_k(-\frac{1}{\Delta z} - 0.05\tilde{s}) - \frac{g}{2}(s_k + s_{k+1})\right] \Big/ (-\frac{1}{\Delta z} + 0.05\tilde{s}).$$

ϕ_{k+1} can now be used in the calculation of ϕ_z at level k.

The upper boundary condition differs from the lower boundary condition. We define an upper boundary at $z = Z_T$ and require that there be no flux of mass, momentum or energy through the boundary. We again define fictitious points outside the boundary.



The boundary condition is imposed by setting

$$(u,v,w,s,\phi)_0 = (u,v,w,s,\phi)_1.$$

With these exterior values the vertical advection terms can be computed in the layer next to the upper boundary.

REFERENCES

Arakawa, A., and V. R. Lamb, *Computational design of the basic dynamical processes of the UCLA General Circulation Model,* Methods in Computational Physics, Vol. 17, Academic Press, pp. 174-265, 1977

Barker, E. H., *Solving for temperature using unnaturally latticed hydrostatic equations,* Monthly Weather Review, Vol. 108, pp.1260–1268, 1980

Berger, M., *Adaptive mesh refinement for hyperbolic partial differential equations.* PhD dissertation, Department of Computer Science, Stanford Univ., California, 1982

Berger, M., *Stability of interfaces with mesh refinement,* Mathematics of Computation, Vol. 45, No. 172, pp. 301-318, 1985

Berger, M., *On conservation at grid interfaces,* Institute for Computer Applications in Science and Engineering, Report No. 84-83, NASA Langley Research Center, 1984

Berger, M., and A. Jameson, *Automatic adaptive grid refinement for the Euler equations* American Institute of Aeronautics and Astronautics Journal, Vol. 23, No. 4, pp. 561–568, 1985

Berger, M., and J. Oliger, *Adaptive mesh refinement for hyperbolic partial differential differential equations,* Journal of Computational Physics, Vol. 53, No. 3, pp. 484–512, March 1984

Browning, G., and Kreiss, H., *Scaling and computation of smooth atmospheric motions,* unpublished manuscript, September, 1985

Browning, G., and Kreiss, H., *Scaling and computation of smooth atmospheric motions,* Lecture Notes in Physics, Vol. 264, Springer-Verlag, NY,NY pp. 128–133, 1986

Browning, G., H. Kreiss, and J. Oliger, *Mesh refinement,* Mathematics of Computation, Vol. 27, No. 121, pp. 29–39, 1973

Caruso, S. C., *Adaptive grid techniques for fluid flow problems,* Phd dissertation, Department of Mechanical Engineering, Stanford Univ., California, 1985

Dannenhoffer, J. F. and J. R. Baron, *Robust grid adaptation for complex transonic flows,* AIAA 24th Aerospace Sciences Meeting Proceedings Paper, 1986

Haltiner, G.J., and R.T. Williams, *Numerical Prediction and Dynamic Meteorology,* second edition, John Wiley and Sons, N.Y., N.Y., 1980

Harrison, E.J., Jr., *Three dimensional numerical simulations of tropical systems using nested finite grids,* Journal of the Atmospheric Sciences, Vol. 30, pp.1528–1543, 1973

Henshaw, W. D., *Part I: The numerical solution of hyperbolic systems of conservation laws, Part II: Composite overlapping grid techniques,* PhD thesis, California Institute of Technology, Pasadena, California, 1985

Holton, J. R., *An Introduction to Dynamic Meteorology,* 2nd edition, Academic Press, 1979

Hoskins, B. J., and F. P. Bretherton, *Atmospheric frontogenesis models: Mathematical formulation and solution,* Journal of the Atmospheric Sciences, Vol. 29, pp. 11–37, 1972

Jones, R. W., *A nested grid for a three-dimensional model of a tropical cyclone,* Journal of the Atmospheric Sciences, Vol. 34, pp.1528–1553, 1977

Kreiss, H. O., *Problems with different time scales for partial differential equations,* Comm. Pure Appl. Math., Vol. 33, pp. 399-440, 1980

Mudrick, S. E., *A numerical study of frontogenesis,* Journal of the Atmospheric Sciences, Vol 31, No. 5, pp. 869–892, 1974

Oliger, J., and A. Sundström, *Theoretical and practical aspects of some initial value problems in fluid dynamics,* SIAM Journal of Applied Mathematics, Vol. 35, pp. 419–446, 1978

Thomée, V., *Stability theory for partial differential operators,* Society for Industrial and Applied Mathematics Review, Vol. 11, pp. 152–195, 1969

Wiin-Nielsen, A., *On the propogation of gravity waves in a hydrostatic, compressible fluid with vertical wind shear,* Tellus, Vol. 17, pp.306–320, 1965

Williams, R. T., *Atmospheric frontogenesis: A numerical experiment,* Journal of the Atmospheric Sciences, Vol. 24, pp. 627–641, 1967