View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Calhoun, Institutional Archive of the Naval Postgraduate School

Calhoun
Institutional Archive of the Naval Postgraduate School

Calhoun: The NPS Institutional Archive

Faculty and Researcher Publications
Faculty and Researcher Publications

2001

A spectral element shallow water model on spherical geodesic grids

Giraldo, F. X.

International Journal for Numerical Methods in Fluids / Volume 35, Issue 8, 869-901
http://hdl.handle.net/10945/25511

DUDLEY
KNOX
LIBRARY

Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2001; **35**: 869–901

A spectral element shallow water model on spherical geodesic grids[†]

Francis X. Giraldo*,[1]

Naval Research Laboratory, Monterey, CA, U.S.A.

SUMMARY

The spectral element method for the two-dimensional shallow water equations on the sphere is presented. The equations are written in conservation form and the domains are discretized using quadrilateral elements obtained from the generalized icosahedral grid introduced previously (Giraldo FX. Lagrange–Galerkin methods on spherical geodesic grids: the shallow water equations. *Journal of Computational Physics* 2000; **160**: 336–368). The equations are written in Cartesian co-ordinates that introduce an additional momentum equation, but the pole singularities disappear. This paper represents a departure from previously published work on solving the shallow water equations on the sphere in that the equations are all written, discretized, and solved in three-dimensional Cartesian space. Because the equations are written in a three-dimensional Cartesian co-ordinate system, the algorithm simplifies into the integration of surface elements on the sphere from the fully three-dimensional equations. A mapping (Song Ch, Wolf JP. The scaled boundary finite element method—alias consistent infinitesimal finite element cell method—for diffusion. *International Journal for Numerical Methods in Engineering* 1999; **45**: 1403–1431) which simplifies these computations is described and is shown to contain the Eulerian version of the method introduced previously by Giraldo (*Journal of Computational Physics* 2000; **160**: 336–368) for the special case of triangular elements. The significance of this mapping is that although the equations are written in Cartesian co-ordinates, the mapping takes into account the curvature of the high-order spectral elements, thereby allowing the elements to lie entirely on the surface of the sphere. In addition, using this mapping simplifies all of the three-dimensional spectral-type finite element surface integrals because any of the typical two-dimensional planar finite element or spectral element basis functions found in any textbook (for example, Huebner *et al. The Finite Element Method for Engineers*. Wiley, New York, 1995; Karniadakis GE, Sherwin SJ. *Spectral/hp Element Methods for CFD*. Oxford University Press, New York, 1999; and Szabó B, Babuška I. *Finite Element Analysis*. Wiley, New York, 1991) can be used. Results for six test cases are presented to confirm the accuracy and stability of the new method. Published in 2001 by John Wiley & Sons, Ltd.

KEY WORDS: Adams–Bashforth; cubic gnomonic grid; finite element method; icosahedral grid; Lagrange multiplier; Runge–Kutta; shallow water equations; spectral element method; spherical geometry; unstructured grid

---

[†] This article is a U.S. Government work and is in the public domain in the U.S.A.

* Correspondence to: Naval Research Laboratory, 7 Grace Hopper Avenue, Stop 2, Building 704-245, Monterey, CA 93943-5502, U.S.A.

[1] E-mail: giraldo@nrlmry.navy.mil

Published in 2001 by John Wiley & Sons, Ltd.

1. INTRODUCTION

The spectral element method combines the benefits of both the spectral and finite element methods. The spectral element method can automatically generate any order polynomial basis function, as in the spectral method, while allowing for the geometrical flexibility enjoyed by the finite element method. This geometric flexibility permits the use of an unstructured grid. The spectral element method may use any type of Jacobi polynomial to define the basis functions, but typically either Chebyshev or Legendre polynomials are used; in this paper, Legendre polynomials are used. The polynomial order of the spectral element method can be generated automatically along with the corresponding numerical integration rule, which makes the method very suitable for constructing general codes. By selecting the numerical integration (quadrature) points to be the collocation points results in a diagonal mass matrix that reduces the storage requirements. The efficiency of the method is further enhanced by not having to define the basis functions explicitly because implicit relations for the inner products of the functions and their derivatives can be defined *a priori* [1]. Since the collocation points are not equi-spaced, staggered grids can be generated automatically by using varying-order polynomials for the different variables (say the pressure and velocity in Navier–Stokes) as is done in Reference [2] for the shallow water equations. This satisfies the Babuška–Brezzi (or inf–sup) condition, thereby avoiding the development of spurious pressure modes. Many researchers have used the spectral element method successfully for advection–diffusion [3], Stokes flow [1], Navier–Stokes [4], and the shallow water equations [2,5,6].

The recent paradigm shift in large-scale computing from vector machines (having few but very powerful processors) to distributed memory machines (having many but less powerful processors) has led atmospheric scientists to explore methods other than the spectral method for solving the governing equations on the sphere. (The shallow water equations have been used customarily to develop new numerical methods for atmospheric models because they exhibit the same wave behavior as the more complex baroclinic equations governing the motion of the atmosphere.) Besides not parallelizing well, the spectral method also suffers from the restriction that the grid be a longitude–latitude grid, which packs too many unnecessary points at the poles. By exploring other classes of methods, researchers are also free to choose other types of grids. For example, in Reference [6] a cubic gnomonic grid is used in conjunction with the spectral element method. In Reference [7] an icosahedral grid is employed with the finite difference method. In Reference [8] a spiral triangular grid similar to the icosahedral grid is used with the weak Lagrange–Galerkin finite element method. (For a complete review of grids for tiling the sphere see Reference [9].) However, many of these new approaches continue to follow in the footsteps of the spectral method by writing the equations in spherical co-ordinates; the only exception is the finite difference method presented in Reference [7], which solves the equations in a co-ordinate invariant form.

Although spherical co-ordinates seem to be the natural choice on the sphere, they present many problems and associated computational costs. As mentioned above, the spectral method on the sphere requires the use of a longitude–latitude Gaussian grid, which introduces too many redundant points around the poles and this situation is exacerbated as the resolution increases. Finite difference, finite element, and spectral element methods suffer from the existence of singularities in the equations at the poles in this co-ordinate system. This difficulty can be circumvented by applying rotation transformations (as is done in Reference [6] from spherical to gnomonic space) or by using Cartesian co-ordinates.

In References [10–13] the equations are solved in Cartesian rather than spherical co-ordinates. In Reference [10] the Lagrange multiplier approach for transforming the shallow water equations from spherical to Cartesian co-ordinates is introduced. This idea is then used to construct a semi-Lagrangian shallow water model but on a longitude–latitude grid, an unnecessary remnant of spectral methods. In Reference [13] the Taylor–Galerkin method in three-dimensional Cartesian space using an icosahedral grid is presented. However, in Reference [13], the work stops short by then mapping the three-dimensional linear triangles to a local two-dimenaional space; an unnecessary step since all the computations can be carried out in three-dimensional space, thereby avoiding any type of mapping to a local space. In References [11,12] we developed a weak Lagrange–Galerkin finite element method in three-dimensional Cartesian space using triangular linear basis functions on an icosahedral grid; this method can only use linear elements, however.

In the current paper, the equations are once again formulated in three-dimensional Cartesian space (as in References [11,12]), thus simplifying the construction of surface integrals directly onto the three-dimensional space. One of the many differences between this work and that presented in References [11,12] is that it permits the use of very high-order quadrilateral spectral elements (up to 32nd order) rather than linear triangular elements. Quadrilaterals had to be adopted over triangles because no C^0 nodal expansion exists for triangles (modal expansions do exist however, see References [14,15]). Some promising ideas for constructing spectral elements on triangles can be found in References [16,17] and we hope to return to this in a future paper.

In order to be able to construct these high-order elements directly onto the three-dimensional Cartesian space, a special mapping had to be developed (this mapping was derived independently by the current author and, long before, by Song and Wolf [18]). In order to test whether the mapping is mathematically correct, the Eulerian method described in References [11,12] was compared against the new mapping for the special case of linear triangular elements; in Appendix A we show that, for this case, the two methods yield equivalent results thereby proving that the mapping is indeed correct.

Writing the equations in Cartesian space and using the mapping described herein allows the discretization of the equations in conservation form; this has beneficial implications for the conservation properties of the scheme and it also yields a form much more suitable for using standard computational fluid dynamic (CFD) techniques. In fact, in this form the shallow water equations appear as a non-homogeneous case of the Euler equations that govern the motion of inviscid compressible flow.

The general triangular icosahedral grid introduced in Reference [12] is used for constructing the higher-order spectral elements on the sphere. Each triangle is divided into three quadrilaterals because we require quadrilateral elements in order to construct a high-order C^0 nodal expansion. The current paper represents a departure from previously published work on solving the shallow water equations on the sphere in that the equations are all written, discretized, and solved in three-dimensional Cartesian space, which simplifies all of the calculations. This simplification is especially useful when dealing with high-order spectral element basis functions on a curved surface (i.e., a sphere).

The method described herein can be used to construct any type of high-order finite element method, including the nodal expansion described here and the modal expansion using triangular or quadrilateral hierarchic basis functions described in References [14,19]. However, this paper only focuses on a spectral-type finite element method (nodal expansion) on quadrilateral grids. In the next section, we describe the governing equations used in this paper, namely, the shallow water equations.

2. SHALLOW WATER EQUATIONS

The two-dimensional spherical shallow water equations in Cartesian conservation form are

$$\frac{\partial}{\partial t}\begin{bmatrix}\varphi\\ \varphi u\\ \varphi v\\ \varphi w\end{bmatrix}+\frac{\partial}{\partial x}\begin{bmatrix}\varphi u\\ \varphi u^2+\frac{1}{2}\varphi^2\\ \varphi uv\\ \varphi uw\end{bmatrix}+\frac{\partial}{\partial y}\begin{bmatrix}\varphi v\\ \varphi uv\\ \varphi v^2+\frac{1}{2}\varphi^2\\ \varphi vw\end{bmatrix}+\frac{\partial}{\partial z}\begin{bmatrix}\varphi w\\ \varphi uw\\ \varphi vw\\ \varphi w^2+\frac{1}{2}\varphi^2\end{bmatrix}$$

$$=\begin{bmatrix}0\\ f\left(\frac{z}{a}\varphi v-\frac{y}{a}\varphi w\right)+\mu x\\ f\left(\frac{x}{a}\varphi w-\frac{z}{a}\varphi u\right)+\mu y\\ f\left(\frac{y}{a}\varphi u-\frac{x}{a}\varphi v\right)+\mu z\end{bmatrix}$$

but note that if we move the pressure terms to the right-hand side, we get

$$\frac{\partial}{\partial t}\begin{bmatrix}\varphi\\ \varphi u\\ \varphi v\\ \varphi w\end{bmatrix}+\frac{\partial}{\partial x}\begin{bmatrix}\varphi u\\ \varphi u^2\\ \varphi uv\\ \varphi uw\end{bmatrix}+\frac{\partial}{\partial y}\begin{bmatrix}\varphi v\\ \varphi uv\\ \varphi v^2\\ \varphi vw\end{bmatrix}+\frac{\partial}{\partial z}\begin{bmatrix}\varphi w\\ \varphi uw\\ \varphi vw\\ \varphi w^2\end{bmatrix}$$

$$=\begin{bmatrix}0\\ -\varphi\frac{\partial\varphi}{\partial x}+f\left(\frac{z}{a}\varphi v-\frac{y}{a}\varphi w\right)+\mu x\\ -\varphi\frac{\partial\varphi}{\partial y}+f\left(\frac{x}{a}\varphi w-\frac{z}{a}\varphi u\right)+\mu y\\ -\varphi\frac{\partial\varphi}{\partial z}+f\left(\frac{y}{a}\varphi u-\frac{x}{a}\varphi v\right)+\mu z\end{bmatrix} \qquad (1)$$

This system can now be written more compactly as

$$\frac{\partial \boldsymbol{\varphi}}{\partial t} + \nabla \cdot (\boldsymbol{\varphi} \mathbf{u}) = \mathbf{S}(\boldsymbol{\varphi}) \tag{2}$$

where

$$\boldsymbol{\varphi} = \begin{bmatrix} \varphi \\ \varphi u \\ \varphi v \\ \varphi w \end{bmatrix}, \qquad \mathbf{u} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}, \qquad \mathbf{S}(\boldsymbol{\varphi}) = \mathbf{F}_P + \mathbf{F}_R + \mathbf{F}_C \tag{3}$$

$$\mathbf{F}_P = -\varphi \nabla \varphi, \qquad \mathbf{F}_R = -f(\mathbf{r} \times \varphi \mathbf{u}), \qquad \mathbf{F}_C = \mu \mathbf{r}$$

and

$$f = \frac{2\Omega z}{a}$$

where Ω and a are the rotation of the earth and its radius respectively. In the above relations $\mathbf{F}_P$ is the force due to pressure, $\mathbf{F}_R$ is the force due to the rotation of the earth (Coriolis), and $\mathbf{F}_C$ is the force required to constrain the fluid particles to remain on the surface of the sphere, and μ is a Lagrange multiplier used to satisfy this condition. This term is best obtained in discrete form. If it is obtained from the continuous equations, then for the discrete equations the particles may no longer be guaranteed to remain on the surface of the sphere.

The equations are solved for the four conservation variables $\boldsymbol{\varphi} = (\varphi, \varphi u, \varphi v, \varphi w)$, where φ denotes the geopotential height and $\mathbf{u}$ the velocity field. Clearly, we could also include other forcing functions and they would then be included in $\mathbf{S}(\boldsymbol{\varphi})$. In the next section, we show how to discretize Equation (2).

3. DISCRETIZATION

3.1. Spatial discretization

Beginning with Equation (2) we can define a spectral element formulation by taking the weak form

$$\int_{\Omega} \psi \left[\frac{\partial \boldsymbol{\varphi}}{\partial t} + \nabla \cdot (\boldsymbol{\varphi} \mathbf{u}) - \mathbf{S}(\boldsymbol{\varphi}) \right] \mathrm{d}\Omega = 0$$

and integrating by parts (Green's theorem) such as

$$\nabla\cdot(\psi\boldsymbol{\varphi}\mathbf{u}) = \psi\nabla\cdot(\boldsymbol{\varphi}\mathbf{u}) + (\boldsymbol{\varphi}\mathbf{u})\cdot\nabla\psi$$

to arrive at

$$\int_{\Omega} \psi \frac{\partial\boldsymbol{\varphi}}{\partial t}\,\mathrm{d}\Omega + \int_{\Gamma} \mathbf{n}\cdot\mathbf{u}\psi\boldsymbol{\varphi}\,\mathrm{d}\Gamma - \int_{\Omega} \nabla\psi\cdot(\boldsymbol{\varphi}\mathbf{u})\,\mathrm{d}\Omega - \int_{\Omega} \psi\mathbf{S}(\boldsymbol{\varphi})\,\mathrm{d}\Omega = 0$$

where it is understood that the integrals with subscripts Ω denote area integrals and those with Γ denote boundary integrals. In the case of flow on a sphere, the second integral vanishes because the basis functions are chosen to be continuous across element interfaces and there are no true boundaries, in other words

$$\int_{\Gamma} \mathbf{n}\cdot\mathbf{u}\psi\boldsymbol{\varphi}\,\mathrm{d}\Gamma = 0$$

and we are left with

$$\int_{\Omega} \psi \frac{\partial\boldsymbol{\varphi}}{\partial t}\,\mathrm{d}\Omega = \int_{\Omega} \nabla\psi\cdot(\boldsymbol{\varphi}\mathbf{u})\,\mathrm{d}\Omega + \int_{\Omega} \psi\mathbf{S}(\boldsymbol{\varphi})\,\mathrm{d}\Omega$$

By letting the variables $\boldsymbol{\varphi}$ be approximated by $\Sigma_{i=1}^{N_P}\,\psi_i\boldsymbol{\varphi}_i$, where N_P represents the total number of collocation points within each element, we can then write relations for the integrals above (the basis functions ψ are described later in this paper). This yields the following integral equation:

$$\int_{\Omega} \psi_i\psi_j\,\mathrm{d}\Omega\,\frac{\partial\boldsymbol{\varphi}_j}{\partial t} = \left(\int_{\Omega} \nabla\psi_i\psi_j\psi_k\,\mathrm{d}\Omega\cdot\mathbf{u}_k\right)\boldsymbol{\varphi}_j + \int_{\Omega} \psi_i\mathbf{S}(\psi_j\boldsymbol{\varphi}_j)\,\mathrm{d}\Omega$$

which can then be written in the matrix form

$$M_{ij}\frac{\partial\boldsymbol{\varphi}_j}{\partial t} = (\mathbf{A}_{ijk}\cdot\mathbf{u}_k)\boldsymbol{\varphi}_j + \mathbf{S}_i(\boldsymbol{\varphi}) \tag{4}$$

where it is understood that i, j, $k = 1, \ldots, N_P$ and so M_{ij} represents an $N_P \times N_P$ matrix, and so on.

3.1.1. Element matrices. Substituting in the forces $\mathbf{S}(\boldsymbol{\varphi})$ from the right-hand side of Equation (1) into Equation (4) allows us to write all of the element matrices as follows:

$$M_{ij} = \int_\Omega \psi_i \psi_j \, \mathrm{d}\Omega \tag{5}$$

$$\mathbf{A}_{ijk} = \int_\Omega \nabla \psi_i \psi_j \psi_k \, \mathrm{d}\Omega \tag{6}$$

and

$$S_i(\boldsymbol{\varphi}) = \begin{bmatrix} 0 \\ -P^x_{ijk}\varphi_k\varphi_j + R^z_{ij}(\varphi v)_j - R^y_{ij}(\varphi w)_j \\ -P^y_{ijk}\varphi_k\varphi_j + R^x_{ij}(\varphi w)_j - R^z_{ij}(\varphi u)_j \\ -P^z_{ijk}\varphi_k\varphi_j + R^y_{ij}(\varphi u)_j - R^x_{ij}(\varphi v)_j \end{bmatrix}$$

where

$$P^x_{ijk} = \int_\Omega \psi_i \psi_j \frac{\partial \psi_k}{\partial x} \, \mathrm{d}\Omega, \qquad P^y_{ijk} = \int_\Omega \psi_i \psi_j \frac{\partial \psi_k}{\partial y} \, \mathrm{d}\Omega, \qquad P^z_{ijk} = \int_\Omega \psi_i \psi_j \frac{\partial \psi_k}{\partial z} \, \mathrm{d}\Omega \tag{7}$$

and

$$R^x_{ij} = \int_\Omega \psi_i \psi_j \psi_k f_k \psi_l \frac{x_l}{a} \, \mathrm{d}\Omega, \qquad R^y_{ij} = \int_\Omega \psi_i \psi_j \psi_k f_k \psi_l \frac{y_l}{a} \, \mathrm{d}\Omega, \qquad R^z_{ij} = \int_\Omega \psi_i \psi_j \psi_k f_k \psi_l \frac{z_l}{a} \, \mathrm{d}\Omega \tag{8}$$

The matrices M, $\mathbf{A}$, P^x, and R^x represent the mass, advection, pressure, and rotation matrices respectively. All of these matrices are independent of the conservation variables $\boldsymbol{\varphi}$ and only need to be computed once because they remain constant throughout the integration. The Lagrange multiplier μ has not been included in the force vector $S_i(\boldsymbol{\varphi})$. This term is actually computed at the discrete level and only after each time integration has been completed. The procedure used to compute this term is explained in the section on the time discretization scheme.

3.1.2. Basis functions. The conservation variables belong to the following space:

$$(\varphi, \varphi u, \varphi v, \varphi w) \in H^1(\Omega)$$

and their test functions, likewise, are contained in the space

$$\psi_i \in H^1(\Omega)$$

In other words, they belong to the set of square integrable functions whose first derivatives are also square integrable (Sobolev space). Note that no boundaries are defined in this space at all, this is because flow along the surface of a sphere has no boundaries. The element basis functions are written in the following form [18]:

$$\psi_i(\xi, \eta, \zeta) = \zeta \hat{\psi}_i(\xi, \eta) \tag{9}$$

where ζ denotes the outward pointing co-ordinate of the surface defined by the shape functions $\hat{\psi}_i(\xi, \eta)$; $\zeta = 0$ gives the center of the sphere and $\zeta = 1$ yields the surface of the sphere. The basis functions given by Equation (9) simplify the construction of high-order methods on the sphere or any other curved surface because the functions $\hat{\psi}$ can be the typical finite element or spectral element basis functions in R^2. Without this type of mapping it would not be possible to construct high-order basis functions on curved geometries in Cartesian space as we have done in this paper.

The shape functions defining a specific surface element are obtained as a tensor product from the one-dimensional Legendre cardinal functions as

$$\hat{\psi}_i(\xi, \eta) = h_j(\xi) h_k(\eta) \tag{10}$$

where $i = 1, \ldots, N_P$, $j, k = 1, \ldots, P$ and $N_P = P^2$. The integer P denotes the number of Legendre–Gauss–Lobatto (LGL) points in each direction (ξ and η), and is equal to $P = p + 1$, where p denotes the polynomial order of the Legendre cardinal functions (see References [1–3,5] for further details on these basis functions).

3.1.3. Integration. In order to integrate the matrices defined by Equations (5)–(8) they must first be transformed from the global R^3 space ($\Omega = \Omega(x, y, z)$) onto the the local R^2 surface element space ($\hat{\Omega} = \hat{\Omega}(\xi, \eta)$).

The Jacobian of the transformation is given by

$$J = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \tag{11}$$

with the inverse Jacobian being

$$J^{-1}=\begin{bmatrix} \dfrac{\partial \xi}{\partial x} & \dfrac{\partial \xi}{\partial y} & \dfrac{\partial \xi}{\partial z} \\ \dfrac{\partial \eta}{\partial x} & \dfrac{\partial \eta}{\partial y} & \dfrac{\partial \eta}{\partial z} \\ \dfrac{\partial \zeta}{\partial x} & \dfrac{\partial \zeta}{\partial y} & \dfrac{\partial \zeta}{\partial z} \end{bmatrix}$$

which in terms of the physical co-ordinates is

$$J^{-1}=\frac{1}{|J|}\begin{bmatrix} \dfrac{1}{\zeta}\left(\dfrac{\partial y}{\partial \eta}z - y\dfrac{\partial z}{\partial \eta}\right) & \dfrac{1}{\zeta}\left(y\dfrac{\partial z}{\partial \xi} - \dfrac{\partial y}{\partial \xi}z\right) & \left(\dfrac{\partial y}{\partial \xi}\dfrac{\partial z}{\partial \eta} - \dfrac{\partial y}{\partial \eta}\dfrac{\partial z}{\partial \xi}\right) \\ \dfrac{1}{\zeta}\left(x\dfrac{\partial z}{\partial \eta} - \dfrac{\partial x}{\partial \eta}z\right) & \dfrac{1}{\zeta}\left(\dfrac{\partial x}{\partial \xi}z - x\dfrac{\partial z}{\partial \xi}\right) & \left(\dfrac{\partial x}{\partial \eta}\dfrac{\partial z}{\partial \xi} - \dfrac{\partial x}{\partial \xi}\dfrac{\partial z}{\partial \eta}\right) \\ \dfrac{1}{\zeta}\left(\dfrac{\partial x}{\partial \eta}y - x\dfrac{\partial y}{\partial \eta}\right) & \dfrac{1}{\zeta}\left(x\dfrac{\partial y}{\partial \xi} - \dfrac{\partial x}{\partial \xi}y\right) & \left(\dfrac{\partial x}{\partial \xi}\dfrac{\partial y}{\partial \eta} - \dfrac{\partial x}{\partial \eta}\dfrac{\partial y}{\partial \xi}\right) \end{bmatrix} \tag{12}$$

The determinant of the Jacobian matrix (11) can be written in the following form:

$$|J| = \frac{\partial \mathbf{x}}{\partial \zeta}\cdot\mathbf{G}$$

where

$$\mathbf{G} = \frac{\partial \mathbf{x}}{\partial \xi}\times\frac{\partial \mathbf{x}}{\partial \eta} \tag{13}$$

and $|\mathbf{G}|$ denotes the magnitude of the normal vector to the surface element $\hat{\Omega}$ in local space [20]. All of these terms are easily obtained from the basis functions $\hat{\psi}$ and the spatial co-ordinates $\mathbf{x}$ from the approximations

$$\mathbf{x} = \sum_{i=1}^{N_P} \zeta\hat{\psi}_i\mathbf{x}_i$$

Having defined the mapping from physical to computational space, we can now integrate the element matrices numerically. The mass, advection, pressure, and rotation matrices become

$$M_{ij} = \int_{\Omega} \psi_i\psi_j \,\mathrm{d}\Omega = \sum_{q=1}^{N_Q} w_q\zeta^2|\mathbf{G}(\xi_q, \eta_q)|\hat{\psi}_i(\xi_q, \eta_q)\hat{\psi}_j(\xi_q, \eta_q)$$

$$A^{\mathbf{x}}_{ijk} = \int_{\Omega} \frac{\partial \psi_i}{\partial \mathbf{x}}\psi_j\psi_k \,\mathrm{d}\Omega = \sum_{q=1}^{N_Q} w_q\zeta^2|\mathbf{G}(\xi_q, \eta_q)|\left(\zeta\frac{\partial \hat{\psi}_i}{\partial \xi}\frac{\partial \xi}{\partial \mathbf{x}} + \zeta\frac{\partial \hat{\psi}_i}{\partial \eta}\frac{\partial \eta}{\partial \mathbf{x}} + \hat{\psi}_i\frac{\partial \zeta}{\partial \mathbf{x}}\right)\hat{\psi}_j(\xi_q, \eta_q)\hat{\psi}_k(\xi_q, \eta_q)$$

$$P_{ijk}^{\mathbf{x}} = \int_{\Omega} \psi_i \psi_j \frac{\partial \psi_k}{\partial \mathbf{x}} \, \mathrm{d}\Omega = \sum_{q=1}^{N_Q} w_q \zeta^2 |\mathbf{G}(\xi_q, \eta_q)| \hat{\psi}_i(\xi_q, \eta_q) \hat{\psi}_j(\xi_q, \eta_q) \left(\zeta \frac{\partial \hat{\psi}_k}{\partial \xi} \frac{\partial \xi}{\partial \mathbf{x}} + \zeta \frac{\partial \hat{\psi}_k}{\partial \eta} \frac{\partial \eta}{\partial \mathbf{x}} + \hat{\psi}_k \frac{\partial \zeta}{\partial \mathbf{x}} \right)$$

$$R_{ijk}^{\mathbf{x}} = \int_{\Omega} \psi_i \psi_j \psi_k \psi_l \frac{\mathbf{x}_l}{a} f_k \, \mathrm{d}\Omega = \sum_{q=1}^{N_Q} w_q \zeta^4 |\mathbf{G}(\xi_q, \eta_q)| \hat{\psi}_i(\xi_q, \eta_q) \hat{\psi}_j(\xi_q, \eta_q) \hat{\psi}_k(\xi_q, \eta_q) f_k \hat{\psi}_l(\xi_q, \eta_q) \frac{\mathbf{x}_l}{a}$$

where

$$\mathbf{A}_{ijk} = \mathbf{A}_{ijk}^{x}\mathbf{i} + \mathbf{A}_{ijk}^{y}\mathbf{j} + \mathbf{A}_{ijk}^{z}\mathbf{k}$$

and **i**, **j**, **k** denote the Cartesian directional vectors. The value of ζ need not be of any concern because at the surface of the sphere it is defined by $\zeta = 1$. The integer N_Q represents the total number of LGL quadrature points required within each element.

The number of quadrature points used are $N_Q = Q^2$, where Q represents the number of LGL quadrature points in each direction (ξ and η). In general, for exact integration of the matrices we require $Q = (cP + 1)/2$ quadrature points, where c is an integer constant denoting the factor of the maximum order matrix. For the spherical shallow water equations $c = 4$, corresponding to the rotation matrices R^x, thereby requiring $Q = (4P + 1)/2$ quadrature points. Numerically, it was found that using $Q = P + 2$ worked extremely well for all of the test cases using any grid. Also, for $P \geq 4$ the accuracy differences are insignificant between inverting the mass matrix in its full form or in its diagonal form. For this reason, the mass matrix on the left-hand side is stored in its diagonal form. This is what is normally done in spectral element methods; the justification is that $Q = P$ is normally used which then yields a diagonal matrix anyway. We are currently using $Q = P + 2$ because it appears to be a bit more stable especially for the longer runs. Using $Q = P$ requires the use of filters (see Reference [6]); in our runs with $Q = P + 2$ no filtering of any kind was required.

3.2. Time discretization

Starting from the spatial discretization given in Equation (4), the resulting system of ordinary differential equations (ODEs) can now be written as

$$\frac{\partial \boldsymbol{\varphi}}{\partial t} = H(\boldsymbol{\varphi})$$

where

$$H(\boldsymbol{\varphi}) = M_{ij}^{-1}[(\mathbf{A}_{ijk} \cdot \mathbf{u}_k)\boldsymbol{\varphi}_j + S_i(\boldsymbol{\varphi})]$$

Recall that the matrix M is a diagonal matrix and so it requires very little storage and does not require the inversion of a matrix. After integrating in time by a general family of Runge–Kutta schemes yields

$$\boldsymbol{\varphi}^{k+1} = \boldsymbol{\varphi}^{n} + \Delta t \beta H(\boldsymbol{\varphi})^{k-1}$$

where

$$\beta = \frac{1}{K-k+1}, \quad k = 1, \ldots, K \quad \text{and} \quad H(\boldsymbol{\varphi})^{0} = H(\boldsymbol{\varphi})^{n}$$

This one-step multi-stage method is used for the first couple of iterations, at which point the more efficient (but slightly less accurate) third-order Adams–Bashforth scheme is used. The time discretization then is

$$\boldsymbol{\varphi}^{n+1} = \boldsymbol{\varphi}^{n} + \frac{\Delta t}{12}[23H(\boldsymbol{\varphi})^{n} - 16H(\boldsymbol{\varphi})^{n-1} + 5H(\boldsymbol{\varphi})^{n-2}]$$

which clearly requires knowing the conservation variables $\boldsymbol{\varphi}$ at the three time levels n, $n-1$ and $n-2$. For this reason, we use the Runge–Kutta scheme for the first two time steps.

The third-order Adams–Bashforth scheme has some advantages and disadvantages over other explicit schemes, such as the third- or fourth-order Runge–Kutta schemes. The main drawback of the Adams–Bashforth scheme is that it requires a very small time step for stability purposes as compared with the Runge–Kutta scheme; however, the main advantage is that this scheme is much faster than the Runge–Kutta scheme. By this reason alone, the Adams–Bashforth scheme is usually selected as the time integration scheme of choice for spectral element models [2,5,6].

Iskandarani [2] and Ma [5] and have shown that the stability of the scheme requires the time step limit to be

$$\Delta t \leq \frac{C}{p^{2}N_{\mathrm{e}}} \tag{14}$$

where p is the order of the polynomial, N_{e} is the number of elements, and C is a proportionality constant. Therefore, for a fixed number of elements, increasing the order of the polynomial by a factor of 2 requires the time step to be decreased by a factor of 4. This relation held for all our numerical experiments and in fact was used to determine the time step size for a given p and N_{e}.

3.2.1. Lagrange multiplier constraint.

3.2.1. Lagrange multiplier constraint. Once the solution has been obtained at the new time step, we need to constrain the fluid particles to remain on the surface of the sphere. We then have for the momentum equations

$$(\varphi\mathbf{u})_{\mathrm{c}}^{n+1} = (\varphi\mathbf{u})_{\mathrm{u}}^{n+1} + \mu\mathbf{r}$$

where the subscripts ‘c’ and ‘u’ denote the constrained and unconstrained values. Taking the scalar product of $\mathbf{r}$, where $\mathbf{r}$ is the co-ordinate vector of each grid point, gives

$$\mathbf{r}\cdot(\varphi\mathbf{u})_{\mathrm{c}}^{n+1}=\mathbf{r}\cdot(\varphi\mathbf{u})_{\mathrm{u}}^{n+1}+\mu a^2 \tag{15}$$

For a fluid particle to remain on the surface of the sphere, the velocity field must be orthogonal to the co-ordinate (radial) vector of its grid point. In other words

$$\mathbf{u}\cdot\mathbf{r}=0$$

This means that the left-hand side term in Equation (15) vanishes, and after rearranging, yields

$$\mu=-\frac{\mathbf{r}\cdot(\varphi\mathbf{u})_{\mathrm{u}}^{n+1}}{a^2}$$

4. ICOSAHEDRAL GRIDS

One of the difficulties of using the typical icosahedral grid used in References [7,13] is that it is restricted to only a few grids because the number of grid points increases by a factor of 4 with each refinement of the grid. In other words, from $q=4$ to 5 (where q is the refinement level) we go from 2562 grid points to 10242 grid points. This is a very limiting constraint when trying to do, say, a convergence study on a computer of modest size. One way around this dilemma is to construct the general icosahedral grid we developed in Reference [12].

We begin with the initial icosahedron having 12 points and 20 equilateral triangular elements. Then we subdivide each triangle of the initial icosahedron by an nth order Lagrange polynomial. Before doing so, however, it is best to map this triangle onto a gnomonic space. The most unbiased mapping is obtained by mapping about the centroid of the icosahedral triangle. (Note that because all of the initial triangles are equilateral then the centroid is equivalent to the circumcenter of the triangle.) Let $(\lambda_{\mathrm{c}}, \theta_{\mathrm{c}})$ be the centroid of the triangle we wish to map. Then, the gnomonic mapping is given by

$$x=\frac{a\cos\theta\sin(\lambda-\lambda_{\mathrm{c}})}{\sin\theta_{\mathrm{c}}\sin\theta+\cos\theta_{\mathrm{c}}\cos\theta\cos(\lambda-\lambda_{\mathrm{c}})},\qquad y=\frac{a[\cos\theta_{\mathrm{c}}\sin\theta-\sin\theta_{\mathrm{c}}\cos\theta\cos(\lambda-\lambda_{\mathrm{c}})]}{\sin\theta_{\mathrm{c}}\sin\theta+\cos\theta_{\mathrm{c}}\cos\theta\cos(\lambda-\lambda_{\mathrm{c}})} \tag{16}$$

which is rather complicated. However, by first applying a rotation mapping whereby in the new co-ordinate system the co-ordinates (λ, θ) are located at $(0, 0)$, then Equation (16) becomes

$$x=a\tan\lambda',\qquad y=a\tan\theta'\sec\lambda' \tag{17}$$

where the rotation mapping is given by

$$\lambda' = \arctan\left[\frac{\cos\theta\,\sin(\lambda-\lambda_{\mathrm{c}})}{\sin\theta_{\mathrm{c}}\sin\theta+\cos\theta_{\mathrm{c}}\cos\theta\,\cos(\lambda-\lambda_{\mathrm{c}})}\right]$$

$$\theta' = \arcsin[\cos\theta_{\mathrm{c}}\sin\theta-\sin\theta_{\mathrm{c}}\cos\theta\,\cos(\lambda-\lambda_{\mathrm{c}})] \tag{18}$$

This approach results in the construction of a general icosahedral grid defined by

$$N_{\mathrm{p}}^{\mathrm{T}} = 10(n-1)^2+20(n-1)+12$$

$$N_{\mathrm{e}}^{\mathrm{T}} = 2(N_{\mathrm{p}}^{\mathrm{T}}-2)$$

$$N_{\mathrm{s}}^{\mathrm{T}} = 3(N_{\mathrm{p}}^{\mathrm{T}}-2)$$

where $N_{\mathrm{p}}^{\mathrm{T}}$, $N_{\mathrm{e}}^{\mathrm{T}}$, and $N_{\mathrm{s}}^{\mathrm{T}}$ denote the number of points, elements, and sides comprising the triangular grid, and n is the order of the Lagrange polynomial used to subdivide the 20 initial triangles of the icosahedron. This grid [12] was shown to be much more general and allows much more flexibility in constructing grids than the typical icosahedral grid used in References [7,13]. Note that if we write $n=2^q$ we recover the typical icosahedral grid.

Once we have constructed the triangular icosahedral grid, we then subdivide each triangular element into three smaller quadrilateral elements. We need quadrilaterals because the high-order nodal basis functions are only defined on quadrilaterals and not on triangles ($\hat{\psi}$ in Equation (9)). Upon dividing the triangles into quadrilaterals we can then construct the spectral collocation points inside each element resulting in a quadrilateral grid with the following properties:

$$N_{\mathrm{p}}^{Q} = 6(N_{\mathrm{p}}^{T}-2)(p)^2+2$$

$$N_{\mathrm{e}}^{Q} = 6(N_{\mathrm{p}}^{T}-2)$$

$$N_{\mathrm{s}}^{Q} = 12(N_{\mathrm{p}}^{T}-2)$$

where N_{p}^{Q}, N_{e}^{Q}, and N_{s}^{Q} denote the number of points, elements, and sides comprising the quadrilateral spectral element grid, and p is the order of the spectral element basis functions. Table I gives the grid dimensions for various values of n and p. Figure 1 shows some possible grid configurations for $n=1$ and $p=1,\ldots,32$.

This, of course, is not the only possible grid that can be used for spectral elements on the sphere. Alternatively, we could use the cubic gnomonic grids employed in References [6,21]. In fact, this grid appears to be a much more natural choice because the grid is constructed by mapping a hexahedron (cube) onto the sphere. Each of the six faces can then be subdivided into as many quadrilaterals as required. The main drawback of this grid is that the quadrilateral elements furthest from the centroid of each face are much smaller than those near the centroid due to the distortion of the gnomonic projection given by Equation (16). Even though we are also using this projection for our icosahedral grid we now have 20 smaller elements to map rather than six larger ones for the cubic gnomonic grid. To illustrate the point, comparing both grids for a comparable number of elements we find that the ratio of maximum to minimum element sizes ranges from 1 to 1.96 for the icosahedral grid and from

Table I. The number of grid points (N_p^Q), elements (N_e^Q), and sides (N_s^Q) for the spectral element icosahedral grid using quadrilaterals.

n	p	$n * p$	N_p^Q	N_e^Q	N_s^Q
4	1	4	962	960	1920
8	1	8	3842	3840	7680
16	1	16	15 362	15 360	30 720
32	1	32	61 442	61 440	122 880
64	1	64	245 762	245 760	491 520
1	4	4	962	60	120
2	4	8	3842	240	480
4	4	16	15 362	960	1920
8	4	32	61 442	3840	7680
16	4	64	245 762	15 360	30 720
1	8	8	3842	60	120
2	8	16	15 362	240	480
4	8	32	61 442	960	1920
8	8	64	245 762	3840	7680
1	16	16	15 362	60	120
2	16	32	61 442	240	480
4	16	64	245 762	960	1920
1	4	4	962	60	120
1	8	8	3842	60	120
1	16	16	15 362	60	120
1	32	32	61 442	60	120
1	64	64	245 762	60	120

The order of the generalized icosahedral grid is given by n and the order of the spectral element basis function is given by p.

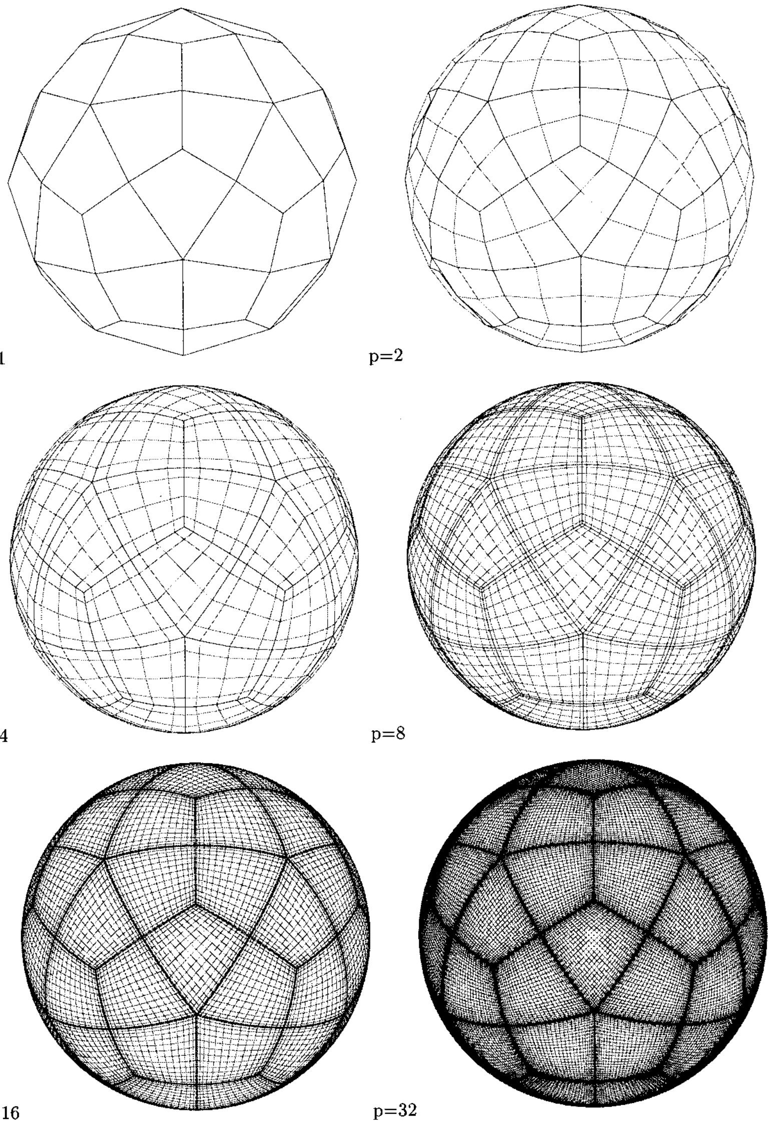
2.20 to 5.14 for the cubic grid (corresponding to $n = 1, \ldots, 64$). Clearly, the icosahedral grid yields a more uniform representation on the sphere which is a highly desirable property for global atmospheric models; the construction of a global atmospheric model is the long term goal of our research.

5. RESULTS

For the numerical experiments, the following three terms are used for judging the performance of the scheme:

the L_2 error norm

$$\|\varphi\|_{L_2} = \sqrt{\frac{\int_\Omega (\varphi_{\text{exact}} - \varphi)^2 \, d\Omega}{\int_\Omega \varphi_{\text{exact}}^2 \, d\Omega}}$$



p=1

p=2

p=4

p=8

p=16

p=32

Figure 1. The grid for $n = 1$ for various values of p, where the elements are defined by the solid lines and the collocation points by the dashed lines.

where $\boldsymbol{\varphi}$ represents the conservation variables, and the following two additional conservation measures:

$$M = \frac{\int_{\Omega} \varphi \, \mathrm{d}\Omega}{\int_{\Omega} \varphi_{\mathrm{exact}} \, \mathrm{d}\Omega}, \qquad E = \frac{\int_{\Omega} \varphi(u^2 + v^2 + w^2) + \varphi^2 \, \mathrm{d}\Omega}{\int_{\Omega} \varphi_{\mathrm{exact}}(u^2_{\mathrm{exact}} + v^2_{\mathrm{exact}} + w^2_{exact}) + \varphi^2_{exact} \, \mathrm{d}\Omega}$$

The L_2 error norm compares the root-mean-square (r.m.s.) per cent error of the numerical and exact solutions, M measures the conservation property of the mass, and E measures the conservation of the total available energy. The ideal scheme should yield an L_2 error norm of zero, and mass and energy measures of one.

Six test cases are used in order to test the algorithm. Test cases 1, 2, 3, 5, and 6 correspond to the test cases given in Reference [22]. Test case 4 is given in Reference [23]. Test case 1 involves the mass equation only, whereas the remainder of the test cases concern the full shallow water equations. In addition, cases 1–3 have analytic solutions and are used to determine the accuracy of the spectral element method quantitatively. Test cases 4–6, on the other hand, do not have analytic solutions and are thus only used to determine the accuracy of the scheme qualitatively.

For all of the test cases studied in this paper, the time steps used were $\Delta t = 864$, 216, 54, and 14 s for the grids $(n = 1)$ $p = 4$, 8, 16, and 32 respectively. Note that the time step size decreases by a factor of 4 for p increasing from 2^m to 2^{m+1} for $m = 2, \ldots, 4$ as given by the Adams–Bashforth stability limit (Equation (14)).

5.1. Case 1: steady state advection

This test case concerns the solid body rotation of a cosine wave. It only tests the mass equation as the velocity field is assumed to remain unchanged throughout the computation. Results are obtained after one full revolution, which corresponds to an integration of 12 days.

By using the mapping from spherical to Cartesian space

$$x = a \cos\theta \cos\lambda$$

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

$$z = a \sin\theta$$

where

$$\lambda = \arctan\left(\frac{y}{x}\right), \qquad \theta = \arcsin\left(\frac{z}{a}\right)$$

we can write the initial conditions in terms of Cartesian co-ordinates. The spherical co-ordinates (λ, θ) correspond to the longitude and latitude. This results in the following velocity field in Cartesian space

$$u = -u_s \sin\lambda - v_s \sin\theta \cos\lambda$$

$$v = +u_s \cos\lambda - v_s \sin\theta \sin\lambda$$

$$w = +v_s \cos\theta$$

where u_s and v_s are the zonal and meridional velocity components in spherical co-ordinates.

A convergence study is shown in Figure 2 for the schemes $p = 1$, $p = 4$, $p = 8$, $p = 16$, and $n = 1$. The L_2 error of the mass is plotted against the product $n * p$. The scheme $p = 1$ represents keeping the order of the polynomial fixed at first-order while increasing the number of elements (increasing n from 4 to 32). Likewise, $p = 4$ represents keeping the order of the polynomial fixed at fourth order while increasing the number of elements (increasing n from 1 to 8). Similarly, $p = 8$ represents keeping the order of the polynomial fixed at eighth order while increasing the number of elements (increasing n from 1 to 4), and $p = 16$ represents keeping the order of the polynomial fixed at sixteenth order while increasing the number of elements (increasing n from 1 to 2). Finally, $n = 1$ represents keeping the number of elements fixed ($n = 1$) but increasing the order of the polynomial (increasing p from 4 to 32). Table I contains the number of grid points and elements for all of the grid combinations of p and n used throughout the paper.

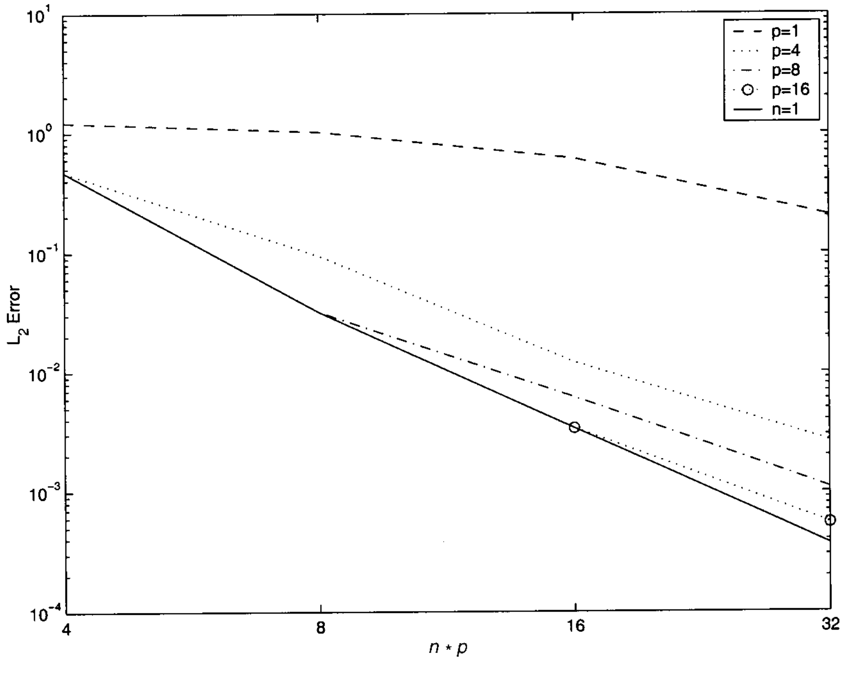


Figure 2. Case 1. Convergence study of the L_2 mass error for various values of $n * p$.

This study illustrates that a better convergence rate is achieved by keeping the number of elements fixed and increasing the order of the polynomial ($n = 1$). The $p = 1$ scheme converges at a rate of 2.15, meaning that by doubling the variable $n*p$ we decrease the error by a factor of 2.15, while the $p = 4$, $p = 8$, $p = 16$, and $n = 1$ schemes converge at rates of 5.85, 6.32, 6.60, and 10 respectively. The $n = 1$ scheme convergences at a spectral (exponential) rate; this is the p-version of the spectral element method. The $p = 1$ scheme gives the slowest convergence rate and it represents using linear elements but increasing the number of elements (increasing n from 4 to 32); this is the h-version of the spectral element method and, in fact, it is identical to a bilinear finite element method. For smooth flows (which this test case represents), the p-version is superior to the h-version but for general problems a combination of both (the h–p method) is optimal. For this reason we have also included the $p = 4$, $p = 8$, and $p = 16$ schemes.

5.2. Case 2: global steady state non-linear zonal geostrophic flow

This case is a steady state solution to the non-linear shallow water equations. The equations are geostrophically balanced and remain so for the duration of the integration. The velocity field thus remains constant throughout the computation. The geopotential height (φ) undergoes a solid body rotation, but since the initial height field is given as a constant band, the solution looks the same throughout the time integration. The velocity field is the same as that used in case 1. All the results reported are for a 5-day integration as suggested in Reference [22].

A convergence study is shown in Figure 3 for the $p = 1$, $p = 4$, and $n = 1$ schemes. The convergence rates for the three schemes $p = 1$, $p = 4$, and $n = 1$, are 3.73, 6.50, and 8.74 respectively. Once again, the p-version of the spectral element method ($n = 1$) yields superior results. The $n = 1$ scheme falls slightly short of exponential convergence but it gives impressive results nonetheless. However, it would appear that the scheme reaches machine precision and hence plateaus for increasing $n*p$ values. For this reason, the $p = 8$ and $p = 16$ schemes were omitted from this plot because the results were virtually indistinguishable from the $n = 1$ results. The convergence rate of the $n = 1$ scheme is higher than exponential for $p \leq 8$ but slows down after 8. In fact, from $p = 4$ to $p = 8$ the scheme is superconvergent with the error decreasing by three orders of magnitude!

5.3. Case 3: steady state non-linear zonal geostrophic flow with compact support

This case is similar to case 2 except that the velocity is zero everywhere except in a very small isolated region. This isolated region, or jet, encapsulates the flow and limits the geopotential height field to remain within a very confined circular region. As in case 2, the results correspond to a 5-day integration.

A convergence study is shown in Figure 4 for the $p = 1$, $p = 4$, $p = 8$, $p = 16$, and $n = 1$ schemes. The convergence rates for the five schemes are 2 for the $p = 1$, $p = 4$, $p = 8$, and $p = 16$ schemes, while it is 3.59 for the $n = 1$ scheme. Although the convergence rates for this test case are low relative to the previous two test cases it does show that even for localized phenomena (such as this test case represents) the $n = 1$ scheme yields the best results, converging at the fastest rate.

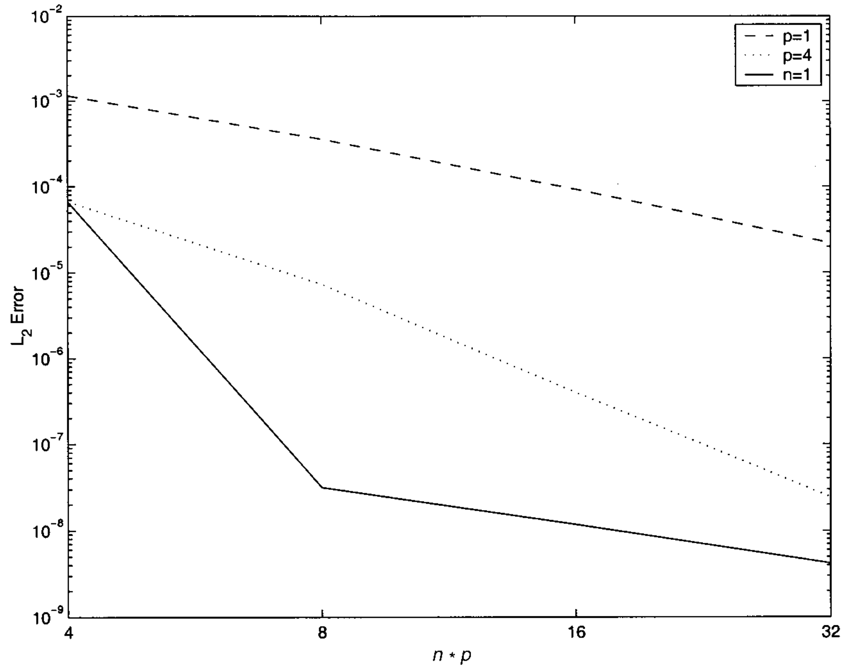


Figure 3. Case 2. Convergence study of the L_2 mass error for various values of $n*p$.

Having performed satisfactorily in these three analytic test cases, we now move to three more test cases without analytic solutions. These test cases shall be used to gauge the qualitative performance of the spectral element method.

5.4. Case 4: dancing high–low waves

This test case comes from Reference [23] and is not an analytic solution to the shallow water equations. The initial geopotential height is comprised of two large waves with the left wave being the low wave and the right wave being the high wave, when viewed from the north pole (see Figures 5–7). The waves rotate clockwise in a dance-fashion so that after 5 days of integration, the high wave is now on the left and the low wave is on the right.

Figures 5–7 show the mass and velocity field after 5 days for the three grid resolutions of $p=4$, $p=8$, and $p=16$ (with $n=1$ for all three grids). The viewpoint in these figures is $(\lambda, \theta)=(0, 90)$, which represents a view from above the north pole. We can see from these three figures that the contours have a similar structure for the three grids but the flow features become better resolved as p is increased. For instance, comparing the mass contour plots for $p=4$ and $p=8$ (Figures 5(a) and 6(a)) we can see that the contour curves have become sharper everywhere. The resolution further increases as we go from $p=8$ to $p=16$ (Figures 6(a) and 7(a)). The increase in resolution is even more apparent with the zonal and meridional velocity

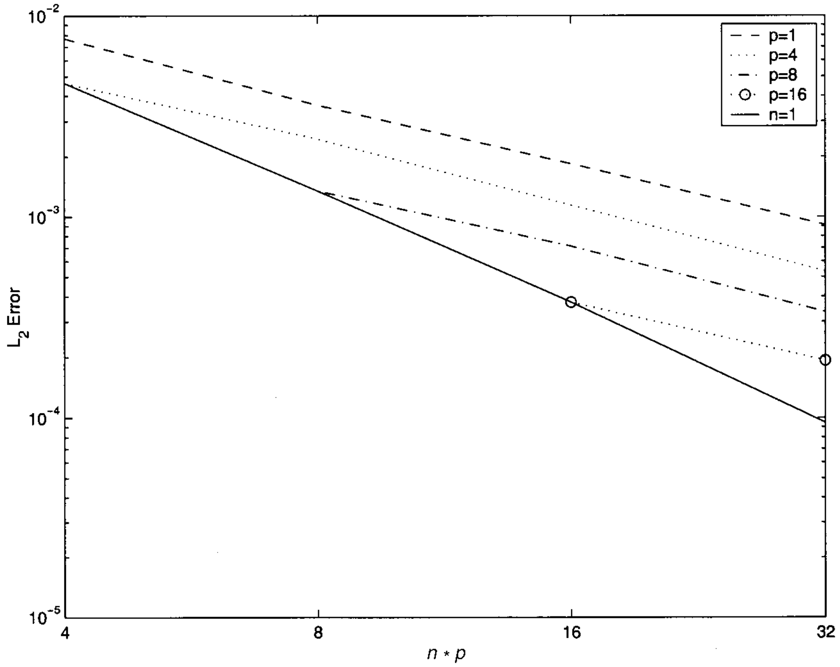


Figure 4. Case 3. Convergence study of the L_2 mass error for various values of $n * p$.

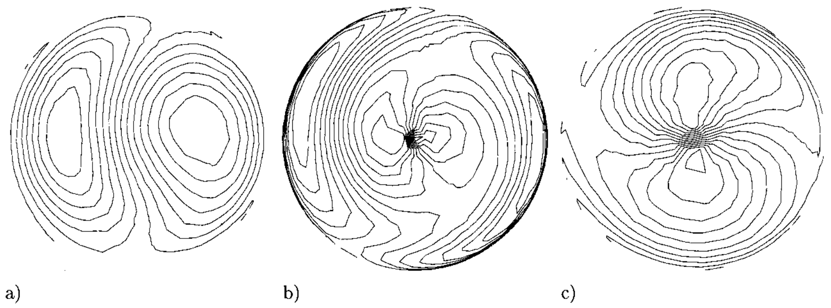


Figure 5. Case 4. The (a) mass, (b) zonal velocity, and (c) meridional velocity on grid $n = 1$, $p = 4$ after 5 days viewed from $(\lambda, \theta) = (0, 90)$.

components. Figure 7(b) and (c) show a much sharper wave structure than those in Figure 5(b) and (c) and Figure 6(b) and (c). The difference in resolution is more apparent near the center of these plots (the north pole) because this is the center of rotation of the two waves; large gradients exist at the center where the low and high waves meet. Clearly, this test case shows

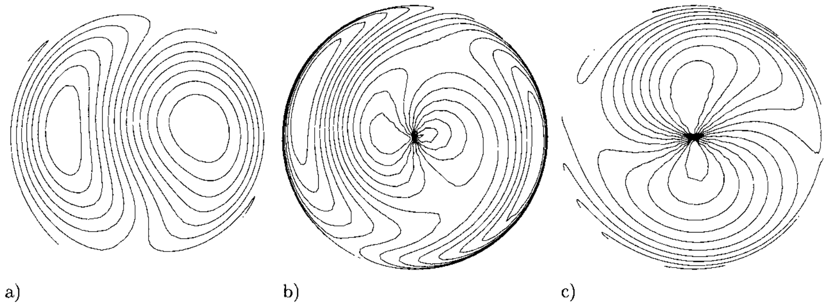


Figure 6. Case 4. The (a) mass, (b) zonal velocity, and (c) meridional velocity on grid $n = 1$, $p = 8$ after 5 days viewed from $(\lambda, \theta) = (0, 90)$.

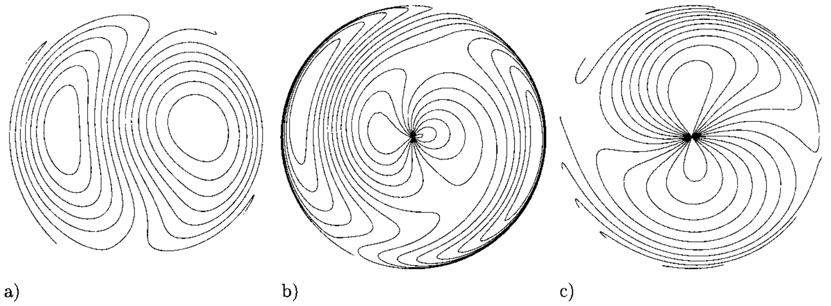


Figure 7. Case 4. The (a) mass, (b) zonal velocity, and (c) meridional velocity on grid $n = 1$, $p = 16$ after 5 days viewed from $(\lambda, \theta) = (0, 90)$.

that the spectral element method is converging towards a wave structure best represented by Figure 7.

We have run this test case for a 15-day integration and the waves remain intact even at this long integration period. However, we only show results after 5 days because the reference solution in Reference [22] only shows results for a 5-day integration. Furthermore, the results illustrated in Figures 5(a), 6(a), and 7(a) are quite similar to Figures 5 and 6 in Reference [23]. The results presented in Reference [23] were obtained by a finite difference semi-Lagrangian method using 7200 points. Because our spectral element method is higher order and we used 15362 points (for $p = 16$), differences between the two solutions are expected.

Because the above contour plots only show a hemisphere, we have plotted the results of the $p = 16$ grid on a longitude–latitude projection in Figure 8, thereby showing the resulting wave structure throughout the entire globe. The longitude–latitude contour plot is shown for a 5-day integration. Note that the longitudinal view in this plot is $\lambda = 180$, whereas in Figures 5–7 it is $\lambda = 0$. For this reason, the high and low waves are reversed in Figure 8. In the

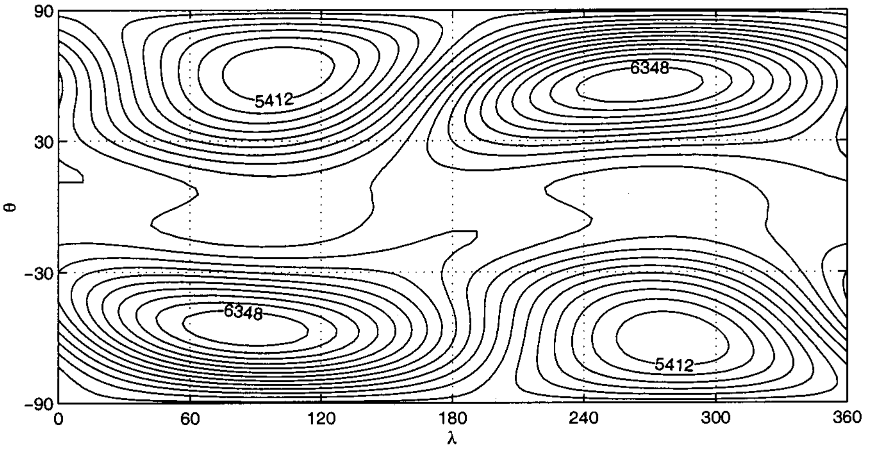


Figure 8. Case 4. Contour plot of the mass for the grid $n = 1$, $p = 16$ after 5 days on a longitude–latitude (λ, θ) projection.

northern hemisphere of this figure, the high wave is clearly seen on the right and the low wave on the left and in the southern hemisphere their positions are reversed. Note that the high waves in the northern and southern hemisphere have the same values. The same holds true for the low waves. This implies that the numerical method and grid are maintaining the initial symmetries of the problem throughout the 5-day integration without introducing any numerical biasing to either hemisphere.

5.5. Case 5: zonal flow over an isolated mountain

This case is similar to case 2 except that now there is a mountain on the sphere. This is the only problem in the test cases studied here which includes topography. The mountain is conical in shape and forces the zonal flow to deflect off the mountain. Due to the zonal flow impinging on the mountain, various wave structures form and propagate around the sphere. The results shown for this test case are for 1-, 5-, and 10-day integrations.

Figures 9–11 show the convergence of the spectral element method for the three different resolutions $p = 4$, $p = 8$, and $p = 16$ (with $n = 1$). The viewpoint used for these plots is $(\lambda, \theta) = (270, 0)$. This viewpoint was chosen because the mountain is located at $(\lambda, \theta) = (270, 30)$ (see Reference [22] for details of this test case). At 1 day, the wave structure generated by the mountain is immediately noticeable in Figures 9(a), 10(a), and 11(a). However, for $p = 16$ (Figure 11(a)) the wave structure is much sharper than for the other two resolutions. For $p = 4$ we get some semblance of the wave structure (Figure 9) but note that the mass contour plots become more jagged at 5 and 10 days. Figure 10 shows that for $p = 8$ the contours are smoother, but they nonetheless lose some smoothness near the north pole and at the equatorial troughs, especially after 10 days. In contrast, the sharpness in resolution achieved by the highest-order scheme, $p = 16$ (Figure 11), is evident at both 5 and 10 days. For

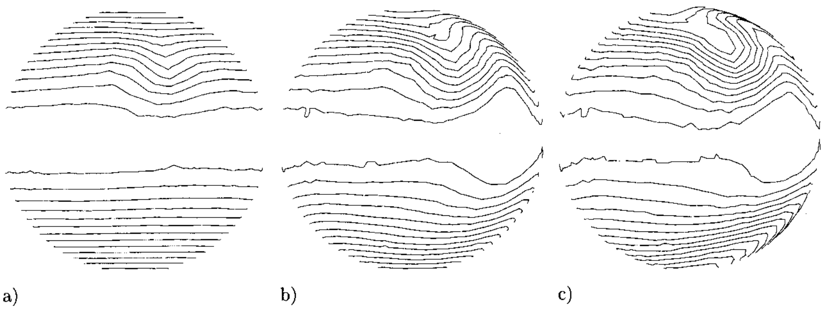


Figure 9. Case 5. The mass at (a) 1, (b) 5, and (c) 10 days on grid $n = 1$, $p = 4$ viewed from $(\lambda, \theta) = (270, 0)$.

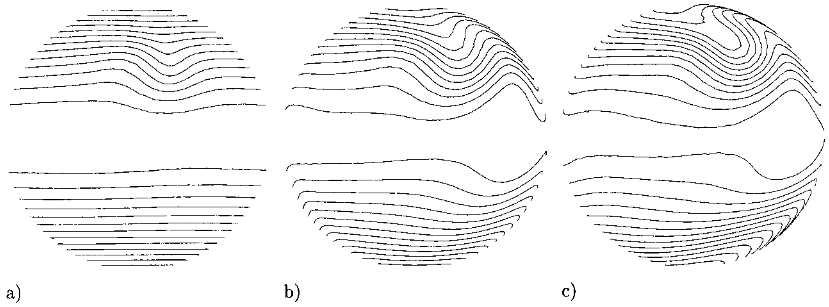


Figure 10. Case 5. The mass at (a) 1, (b) 5, and (c) 10 days on grid $n = 1$, $p = 8$ viewed from $(\lambda, \theta) = (270, 0)$.

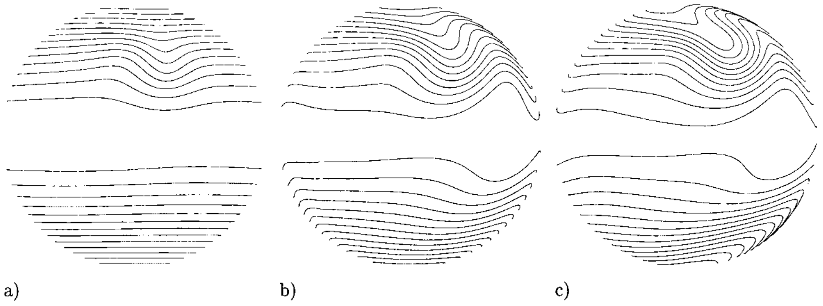


Figure 11. Case 5. The mass at (a) 1, (b) 5, and (c) 10 days on grid $n = 1$, $p = 16$ viewed from $(\lambda, \theta) = (270, 0)$.

example, in the vicinity of the mountain, we observe the contours to be quite sharp. In addition, the equatorial troughs are well defined and the north polar circulation triggered by the mountain is quite smooth. It should be mentioned that no type of smoothing has been used for any of the results illustrated—either for plotting or for damping the high-order wave numbers of the numerical scheme. The smoothness observed in the contour plots is a result of the increased grid resolution; in these cases, obtained by increasing p only (the polynomial order of the spectral element basis functions).

Figure 12 shows the contour plot of the mass on a longitude–latitude projection for $p = 16$ for a 10-day integration. These results show the same wave structure given in References [7,24]; it should be mentioned that the current results have been shifted 180° in order to compare against their results because in the current case the mountain is located at $\lambda = 270$ (as suggested in Reference [22]), whereas in theirs it is stationed at $\lambda = 90$. In fact, the contours in Figure 12 match those in figure 5.1c of Reference [24] exactly. The results presented in Reference [24] were obtained using a spectral method with a resolution of T213, consisting of 204800 points. Our results were obtained with 15362 points ($p = 16$ with $n = 1$).

5.6. Case 6: Rossby–Haurwitz wave

Although Rossby–Haurwitz waves are not analytic solutions to the shallow water equations, they have become a *de facto* test case. In a non-divergent barotropic model, the initial geopotential height field undergoes a solid body rotation in a counterclockwise direction (when viewed from the north pole). The angular velocity is given by

$$v = \frac{R(3+R)\omega - 2\Omega}{(1+R)(2+R)}$$

where $R = 4$ is the wavenumber. Results are reported for 1- and 7-day integrations.

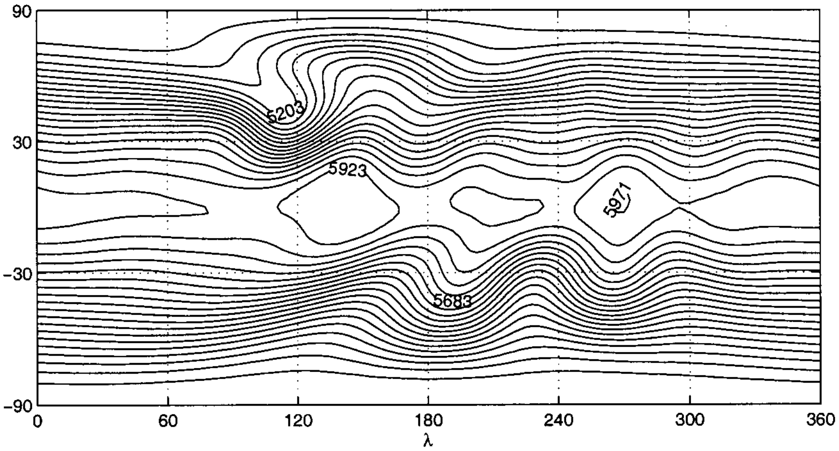


Figure 12. Case 5. Contour plot of the mass for the grid $n = 1$, $p = 16$ after 10 days on a longitude–latitude (λ, θ) projection.

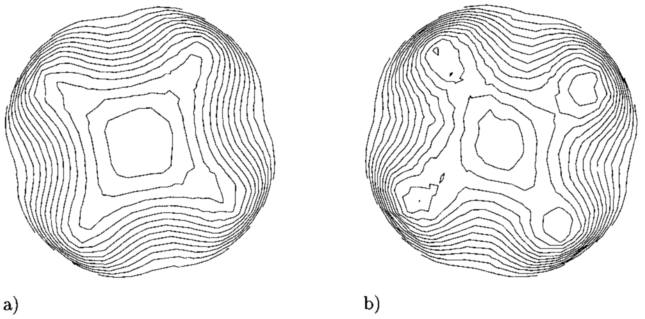


Figure 13. Case 6. The mass at (a) 1 and (b) 7 days on grid $n = 1$, $p = 4$ viewed from $(\lambda, \theta) = (0, 90)$.

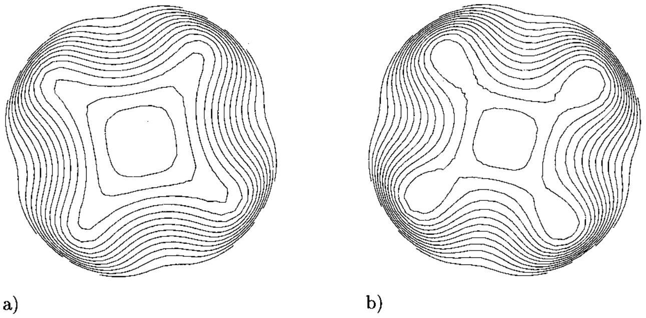


Figure 14. Case 6. The mass at (a) 1 and (b) 7 days on grid $n = 1$, $p = 8$ viewed from $(\lambda, \theta) = (0, 90)$.

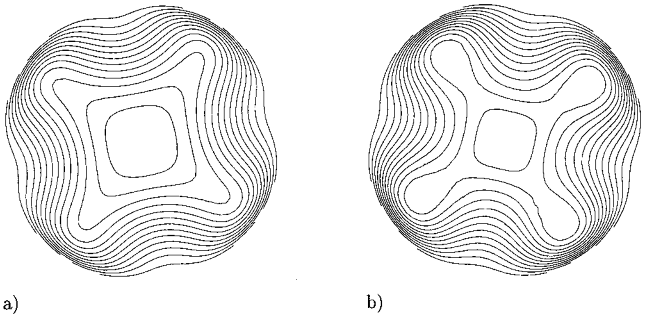


Figure 15. Case 6. The mass at (a) 1 and (b) 7 days on grid $n = 1$, $p = 16$ viewed from $(\lambda, \theta) = (0, 90)$.

Figures 13–15 show the mass contours for the three different grids from the viewpoint $(\lambda, \theta)=(0, 90)$. This particular viewpoint was selected because from this vantage point, the waves undergo a counterclockwise solid body rotation, which then simplifies comparisons between different resolution grids. After 1 day, the wave is completely intact for all three resolutions, but as the resolution is increased a sharper wave image is obtained. At 7 days, the $p=4$ grid begins to dissolve the wave structure, whereas it remains completely intact for the $p=8$ and $p=16$ grids. In addition, the $p=16$ grid gives a sharper image than the $p=8$ grid.

In Figure 16 we have plotted the mass contours on a longitude–latitude projection for the grid $p=16$ after 7 days in order to show the full wave structure throughout the sphere. The results in Figure 16 match those in figure 5.5c of Reference [24] exactly. Once again, the results given in Reference [24] were obtained using a spectral method with a resolution of T213, which is comprised of 204800 points. In comparison, our $p=16$ grid (with $n=1$) contains 15362 points. Note that the solution after 7 days is completely symmetrical, thereby confirming that the scheme has conserved the symmetry of the initial conditions.

5.7. Summary of results

The spectral element method on icosahedral grids performed extremely well in all six test cases. By discretizing the conservation form of the equations with the spectral element method we were able to conserve mass (M) exactly for all six test cases. The total energy (E) was conserved exactly for cases 1–3. For cases 4–6 the spectral element method lost some of its initial energy, with the worst case representing only a very small loss (0.1 per cent for a 15-day integration).

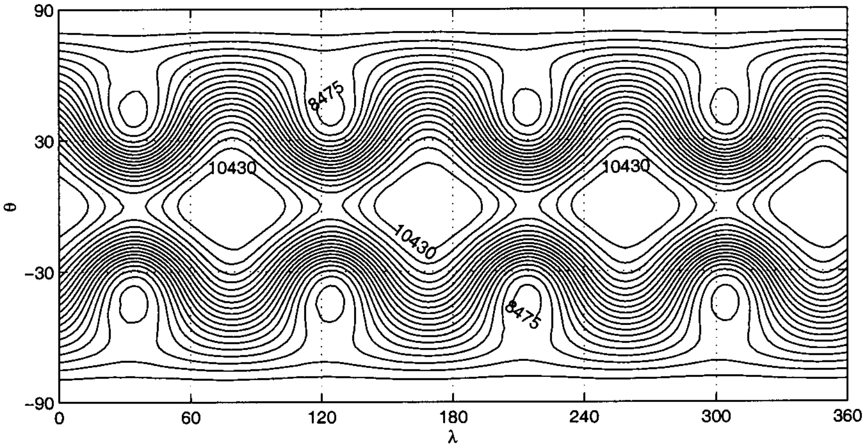


Figure 16. Case 6. Contour plot of the mass for the grid $n=1$, $p=16$ after 7 days on a longitude–latitude (λ, θ) projection.

The results for all six test cases show that the spectral element discretization of the Cartesian conservation form of the shallow water equations on icosahedral grids is extremely accurate. However, these results only prove without a doubt that the method is stable for at least 15 days. Case 1 is clearly stable for any length of time because it only involves the mass equation. Cases 2–6 were run up to 15 days without any instabilities arising.

Currently, the numerical algorithm does not use any type of filtering. Spectral element models typically use some kind of filtering because this method, like the spectral method, is not immune to aliasing errors that arise from the nonlinear terms. It is possible that the higher-order schemes ($p > 8$) of our spectral element method would require some form of filtering in order to suppress the growth of spurious modes for very long time integrations. Further testing of the model should reveal whether or not filtering is required. If so, then we expect to explore *ad hoc* spectral element filters like those presented in References [6,25].

5.8. Computational cost

Figure 17 summarizes the computational cost of using the spectral element method on icosahedral grids for the $p = 1$, $p = 4$, $p = 8$, $p = 16$, and $n = 1$ schemes. These results were obtained using case 3 but are generally representative of our spectral element method. These

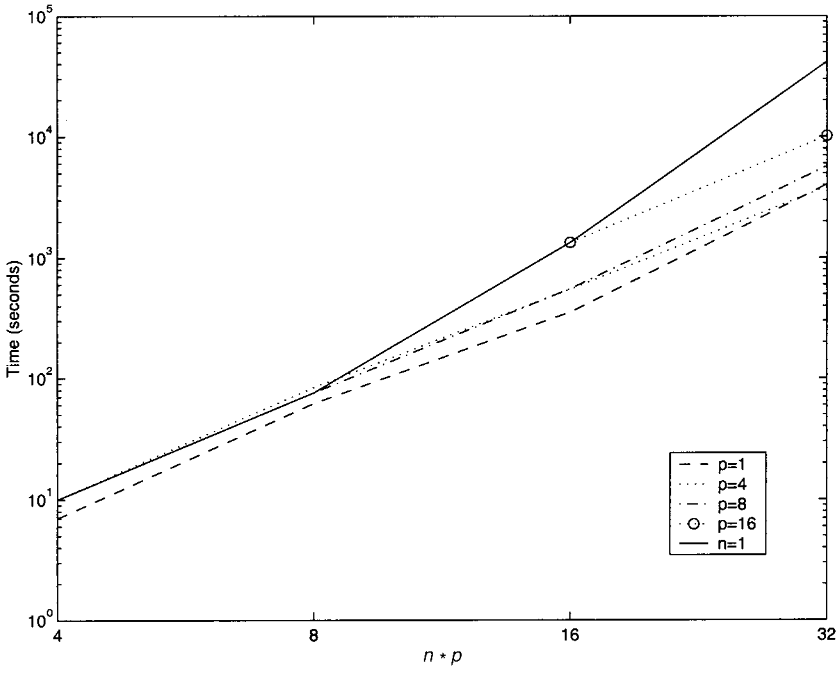


Figure 17. The total computational time incurred by the spectral element method as a function of grid parameter $n * p$ for case 3 for a 5-day integration.

results were obtained on a DEC Alpha 8400 (EV5 CPU chip) with a clock speed of 300 MHz running six threads (multi-threading) using Open MP.

It is obvious from looking at this figure that the h-version of our method ($p=1$) costs the least while the p-version ($n=1$) costs the most. The reason for this is due to the spectral element method requiring $O(n^2p^3)$ operations per time step. Since in the h-method we increase n while keeping p fixed, the number of operations increases by powers of 2, while in the p-method we increase p and hold n constant thereby increasing the number of operations by powers of 3.

This figure shows that the cost of running the $p=4$ and $p=1$ methods are almost identical. In fact, increasing p from 1 to 8 only increases the cost by a factor of 1.5. Increasing p to 16 increases the cost by 2.5. Finally, using the $n=1$ method clearly costs much more, in fact almost an order of magnitude more. These numbers are for $n*p=32$ in Figure 17.

So the issue is to determine what is the best choice of n and p which yields the best results while doing so in the most efficient manner possible. Revisiting Figure 4 we see (again for $n*p=32$) that going from $p=1$ to $p=8$ decreases the error by a factor of 3 while only increasing the cost by a factor of 1.5. Increasing p to 16 reduces the error by a factor of 5 but only increasing the cost by a factor of 2.5. Using the $n=1$ scheme (i.e., $p=32$) decreases the error by an order of magnitude. Of course in this case the cost has also increased by an order of magnitude. Therefore, the optimal range of p must be somewhere between 8 and 32 but we can take the more conservative range of 8–16.

Looking at it another way, we can argue quite convincingly that the high-order method ($p\geq 8$) is much more efficient than the low-order method ($p=1$). In Figure 4 the p value for the $n=1$ curve required to yield the same error as that for the $p=1$ curve for $n*p=32$ is approximately $p=10$ (for an error of 10^{-3}). From Figure 17 the $n=1$ scheme with $p=10$ costs an order of magnitude less than the $p=1$ scheme with $n=32$ (2×10^2 versus 4×10^3 s). Therefore, the high-order methods give not only increased accuracy but increased efficiency as well.

The algorithm currently does not scale linearly with increasing p. Certainly for $p=32$ the method returns as much accuracy as it costs. Going beyond this value will incur even higher costs. However, larger values of p (>32) may become practical if the algorithm can be streamlined and massively parallel architectures exploited. We are currently running an Open MP (multi-threading) version of the model but we hope to implement a message passing interface (MPI) version in the near future. In fact, due to the explicit time differencing currently used it should be trivial to develop an MPI version.

6. CONCLUSIONS

The results obtained for all six problems show that the spectral element method on icosahedral grids offers a viable method for solving the spherical shallow water equations in Cartesian conservation form. Rather than solving the equations in spherical co-ordinates, the equations are transformed to Cartesian co-ordinates, which have two advantages over spherical co-ordinates: the equations no longer contain singularities at the poles, and the primitive form of the equations (momentum form) can be written in conservation form. The conservation form of the equations are the natural form of the equations because they arise from the conservation laws for mass, momentum, and energy. For this reason, this form conserves all of these quantities better than the non-conservation form and has been observed to give better results than the non-conservation form, particularly in the presence of strong gradients [26]. Our results show that mass and energy were conserved up to 99.9 per cent.

By mapping the three-dimensional Cartesian physical space to a local computational surface space we were able to construct high-order spectral-type finite elements on icosahedral grids on the sphere. The high-order basis functions used take into account the curvature of the element as it lies entirely on the sphere; in fact, the surface functions are the typical two-dimensional planar basis functions. As a result, although we discretized the equations in three-dimensional Cartesian space, the problem then reduced to the integration of high-order surface element integrals. In addition, a new spectral icosahedral grid has been introduced which offers many possible grid configurations because it is based on the generalized icosahedral grid introduced previously by the author [12]. The ideas presented in this paper are generalizable towards the construction of a parallel baroclinic model (i.e., fully three-dimensional global atmospheric model), which is the goal of our current research.

ACKNOWLEDGMENTS

The author would like to thank Bob Ribando (University of Virginia) for providing him a contouring algorithm which provided the starting point nearly a decade ago for a plotting package that has been developed by the author. This package was used to plot Figures 1 and 5–15. The author would also like to thank Jeanne Kinney (Harvard University) for reading over the manuscript. This work was supported by the Office of Naval Research through program element PE-0602435N.

APPENDIX A. MAPPING TO COMPUTATIONAL SPACE

In this appendix we show that the method described in this paper contains the Eulerian method described in References [11,12] for the special case of linear triangular elements. To summarize the method introduced in Reference [12], we give the linear basis functions on a triangle in three-dimensional Cartesian space as

$$\tilde{\psi}_i = \frac{1}{|\tilde{J}|}(a_i x + b_i y + c_i z) \tag{19}$$

where

$$|\tilde{J}| = \mathbf{x}_1 \cdot (\mathbf{x}_2 \times \mathbf{x}_3) \tag{20}$$

and

$$a_i = y_j z_k - y_k z_j, \qquad b_i = x_k z_j - x_j z_k, \qquad c_i = x_j y_k - x_k y_j$$

The indices i, j, $k = 1, \ldots, 3$ are the vertices of the triangle and are cycled in the following order:

$$\begin{tikzcd}
 & k \arrow[dl] & \\
i \arrow[rr] & & j \arrow[ul]
\end{tikzcd}$$

and the magnitude of the normal vector to the surface element is given by

$$|\tilde{\mathbf{G}}| = |(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)|$$

In order to show that both methods are equivalent on linear triangular elements, we need only show that the matrices given by Equations (5)–(8) are identical for both methods. Starting with the simplest of these, let us begin with Equation (5). In the case of linear elements, the basis functions in local space $\hat{\Omega}$ are

$$\hat{\psi}_i = \begin{bmatrix} 1 - \xi - \eta \\ \xi \\ \eta \end{bmatrix} \tag{21}$$

If we let $\xi = \tilde{\psi}_2$ and $\eta = \tilde{\psi}_3$ then we could write $\tilde{\psi}_1 = 1 - \xi - \eta$, thereby yielding the exact same basis functions for both methods. In addition by substituting Equation (21) into Equation (13) we get

$$|\mathbf{G}| = \left| \frac{\partial \mathbf{x}}{\partial \xi} \times \frac{\partial \mathbf{x}}{\partial \eta} \right| = |(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)|$$

thereby showing that the mass matrix M is identical for both methods. This also means that any matrix not having a derivative will also be identical, such as the rotation matrices $R^{\mathbf{x}}$. Next we show that the derivatives are equivalent in both methods.

From Equation (9) and the chain rule, we can write the x derivative (where the y and z derivatives would result in similar forms) of the basis functions as

$$\frac{\partial \psi_i}{\partial x} = \zeta \frac{\partial \hat{\psi}_i}{\partial \xi} \frac{\partial \xi}{\partial x} + \zeta \frac{\partial \hat{\psi}_i}{\partial \eta} \frac{\partial \eta}{\partial x} + \hat{\psi}_i \frac{\partial \zeta}{\partial x}$$

which, after substituting Equation (21) gives

$$\frac{\partial \psi_i}{\partial x} = \begin{bmatrix} -\zeta \dfrac{\partial \xi}{\partial x} - \zeta \dfrac{\partial \eta}{\partial x} + \hat{\psi}_i \dfrac{\partial \zeta}{\partial x} \\ \zeta \dfrac{\partial \xi}{\partial x} + \hat{\psi}_i \dfrac{\partial \zeta}{\partial x} \\ \zeta \dfrac{\partial \eta}{\partial x} + \hat{\psi}_i \dfrac{\partial \zeta}{\partial x} \end{bmatrix}$$

The metric terms $\partial\xi/\partial x$ from Equation (12) are

$$\frac{\partial \xi}{\partial x} = \frac{1}{|J|}[(y_3 - y_1)z - y(z_3 - z_1)]$$

$$\frac{\partial \eta}{\partial x} = \frac{1}{|J|}[y(z_2 - z_1) - (y_2 - y_1)z]$$

$$\frac{\partial \zeta}{\partial x} = \frac{1}{|J|}[(y_2 - y_1)(z_3 - z_1) - (y_3 - y_1)(z_2 - z_1)]$$

Substituting these metric terms gives

$$\frac{\partial \psi_1}{\partial x} = \frac{1}{|J|}[-(y_3 - y_1)z + y(z_3 - z_1) - y(z_2 - z_1) + (y_2 - y_1)z + \hat{\psi}_1(y_2 - y_1)(z_3 - z_1) \\ - \hat{\psi}_1(y_3 - y_1)(z_2 - z_1)]$$

where $\zeta = 1$ has been used. After cancelling terms we get

$$\frac{\partial \psi_1}{\partial x} = \frac{1}{|J|}[(y_2 - y_3)z + y(z_3 - z_2) + \hat{\psi}_1(y_2 - y_1)(z_3 - z_1) - \hat{\psi}_1(y_3 - y_1)(z_2 - z_1)]$$

Recall that we can expand the co-ordinates in terms of the basis function approximation

$$\mathbf{x} = \sum_{i=1}^{3} \zeta\hat{\psi}_i \mathbf{x}_i$$

Including this approximation for y and z above and cancelling like terms yields

$$\frac{\partial \psi_1}{\partial x} = \frac{1}{|J|}[(\hat{\psi}_1 + \hat{\psi}_2 + \hat{\psi}_3)(y_2 z_3 - y_3 z_2)]$$

However, the sum of the basis functions (since they are cardinal functions) are equal to 1, thereby giving

$$\frac{\partial \psi_1}{\partial x} = \frac{1}{|J|}(y_2 z_3 - y_3 z_2)$$

From Equation (19) it is immediately obvious that

$$\frac{\partial \tilde{\psi}_1}{\partial x} = \frac{1}{|\tilde{J}|}(y_2 z_3 - y_3 z_2)$$

and so we must now show that $|J| = |\tilde{J}|$. (Note that the same procedure for the rest of the basis functions results in similar forms.) From Equation (11) we can write the Jacobian in $\hat{\Omega}$ as

$$|J| = \frac{\partial \mathbf{x}}{\partial \zeta} \cdot \left(\frac{\partial \mathbf{x}}{\partial \xi} \times \frac{\partial \mathbf{x}}{\partial \eta} \right)$$

which, after substituting the approximation for $\mathbf{x}$ yields

$$J = (\hat{\psi}_1 \mathbf{x}_1 + \hat{\psi}_2 \mathbf{x}_2 + \hat{\psi}_3 \mathbf{x}_3) \cdot [(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)]$$

and expanding the vector product terms gives

$$J = (\hat{\psi}_1 \mathbf{x}_1 + \hat{\psi}_2 \mathbf{x}_2 + \hat{\psi}_3 \mathbf{x}_3) \cdot [\mathbf{x}_2 \times \mathbf{x}_3 - \mathbf{x}_2 \times \mathbf{x}_1 - \mathbf{x}_1 \times \mathbf{x}_3]$$

Next, expanding the scalar product gives

$$J = \hat{\psi}_1 \mathbf{x}_1 \cdot (\mathbf{x}_2 \times \mathbf{x}_3) - \hat{\psi}_2 \mathbf{x}_2 \cdot (\mathbf{x}_1 \times \mathbf{x}_3) - \hat{\psi}_3 \mathbf{x}_3 \cdot (\mathbf{x}_2 \times \mathbf{x}_1)$$

which can be rewritten as

$$J = (\hat{\psi}_1 + \hat{\psi}_2 + \hat{\psi}_3) \mathbf{x}_1 \cdot (\mathbf{x}_2 \times \mathbf{x}_3)$$

Since the sum of the basis functions equals 1, then we get an identical expression to Equation (20). Therefore, any matrix containing a derivative is identical for both methods, which includes the remaining two matrices $\mathbf{A}$ and P^{x}; therefore the Eulerian method given in Reference [12] is contained in the method described here.

REFERENCES

1. Schumack MR, Schultz WW, Boyd JP. Spectral method solution of the Stokes equations on nonstaggered grids. *Journal of Computational Physics* 1991; **94**: 30–58.
2. Iskandarani M, Haidvogel DB, Boyd JP. Staggered spectral element model with application to the oceanic shallow water equations. *International Journal for Numerical Methods in Fluids* 1995; **20**: 393–414.
3. Giraldo FX. The Lagrange–Galerkin spectral element method on unstructured quadrilateral grids. *Journal of Computational Physics* 1998; **147**: 114–146.
4. Henderson RD, Karniadakis GE. Unstructured spectral element methods for simulation of turbulent flows. *Journal of Computational Physics* 1995; **122**: 191–217.

5. Ma H. A spectral element basin model for the shallow water equations. *Journal of Computational Physics* 1993; **109**: 133–149.
6. Taylor M, Tribbia J, Iskandarani M. The spectral element method for the shallow water equations on the sphere. *Journal of Computational Physics* 1997; **130**: 92–108.
7. Heikes R, Randall DA. Numerical integration of the shallow water equations on a twisted icosahedral grid. Part I: Basic design and results of tests. *Monthly Weather Review* 1995; **123**: 1862–1880.
8. Chukapalli G. Weather and climate numerical algorithms: a unified approach to an efficient, parallel implementation. PhD, University of Toronto, 1997.
9. Purser RJ. Non-standard grids. In Proceedings of the European Center for Medium-Range Weather Forecasting Annual Seminar, 7–11 September 1998.
10. Côté J. A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere. *Quarterly Journal of the Royal Meteorological Society* 1988; **114**: 1347–1352.
11. Giraldo FX. Lagrange–Galerkin methods on spherical geodesic grids. *Journal of Computational Physics* 1997; **136**: 197–213.
12. Giraldo FX. Lagrange–Galerkin methods on spherical geodesic grids: the shallow water equations. *Journal of Computational Physics* 2000; **160**: 336–368.
13. Priestley A. The Taylor–Galerkin method for the shallow-water equations on the sphere. *Monthly Weather Review* 1992; **120**: 3003–3015.
14. Dubiner M. Spectral methods on triangles and other domains. *Journal of Scientific Computing* 1993; **6**: 345–390.
15. Sherwin SJ, Karniadakis GE. A triangular spectral element method; applications to the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1995; **123**: 189–229.
16. Hesthaven JS, Gottlieb D. Stable spectral methods for conservation laws on triangles with unstructured grids. *Computer Methods in Applied Mechanics and Engineering* 1999; **175**: 361–381.
17. Taylor MA, Wingate BA. Fekete collocation points for triangular spectral elements. *SIAM Journal of Numerical Analysis* 2001 (in press).
18. Song Ch, Wolf JP. The scaled boundary finite element method—alias consistent infinitesimal finite element cell method—for diffusion. *International Journal for Numerical Methods in Engineering* 1999; **45**: 1403–1431.
19. Szabó B, Babuška I. *Finite Element Analysis*. Wiley: New York, 1991; 95–109.
20. Brebbia CA, Dominguez J. *Boundary Elements An Introductory Course*. McGraw-Hill: San Francisco, 1989; 114–118.
21. McGregor JL. Semi-Lagrangian advection on conformal cubic grids. *Monthly Weather Review* 1996; **124**: 1311–1322.
22. Williamson DL, Drake JB, Hack JJ, Jakob R, Swarztrauber PN. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics* 1992; **102**: 211–224.
23. McDonald A, Bates JR. Semi-Lagrangian integration of a gridpoint shallow water model on the sphere. *Monthly Weather Review* 1989; **117**: 130–137.
24. Jakob-Chien R, Hack JJ, Williamson DL. Spectral transform solutions to the shallow water test set. *Journal of Computational Physics* 1995; **119**: 164–187.
25. Levin J, Iskandarani M, Haidvogel D. A spectral filtering procedure for eddy-resolving simulations with the spectral element ocean model. *Journal of Computational Physics* 1997; **137**: 130–154.
26. Hirsch C. *Numerical Computation of Internal and External Flows*. Wiley: New York, 1991; vol. I, p. 240 and vol. II, pp. 132–144.