Third-Order Finite-Difference Schemes on Icosahedral-Type Grids on the Sphere

J. STEPPELER AND P. RÍPODAS
DWD, Offenbach, Germany

B. JONKHEID
KNMI, De Bilt, Netherlands

S. THOMAS
NCAR, Boulder, Colorado

(Manuscript received 22 February 2007, in final form 11 October 2007)

ABSTRACT

A practical method is proposed to achieve high-order finite-difference schemes on grids that are quasi-homogeneous on the sphere. A family of grids is used that are characterized by the parameter NP, which can take on values of 3, 4, and 5, etc. The parameter NP is the number of grid patches meeting at the Poles. For NP = 3 the cube sphere grid is obtained and for NP = 5 the icosahedron is obtained. While the grid construction method is valid for all values of NP, the tests performed in this paper concern only the case NP = 5 (i.e., the icosahedron). For each of the rhomboidal patches, the grid is created by connecting points on opposing sides of the rhomboid by great circles. This offers the possibility to obtain derivatives for a line of grid points along a great circle in the classical way. Therefore, it becomes possible to use well-known spatial discretizations from limited-area models. Local models can be transferred to the sphere with rather limited effort. The method was tested using the fourth-order Runge–Kutta integration method and fourth-order spatial differencing. At patch limits, boundary values are obtained using third-order serendipity interpolation, giving the scheme an overall space–time accuracy of 3. The serendipity interpolation is quite efficient. Third-order interpolation in two dimensions is achieved by a set of linear interpolations and a number of function evaluations. All coefficients can be precomputed. The third-order convergence is demonstrated by numerical experiments using Williamson’s test cases 2 and 6.

1. Introduction

Many models for global forecasts or climate simulation use the spectral method or the uniform latitude–longitude grid. For a discussion of the potential drawbacks of these methods see Taylor et al. (1997). For the spectral method, these include the Gibbs phenomenon, reduced flexibility in designing adaptive grids with two-way interaction, limitations in using vertical coordinates (such as z coordinate or the Mesingers step approach), and problems with the Legendre transform at high resolutions. The latitude–longitude grid has the problem of the pole singularity.

There have been many attempts to design methods using a more uniform grid, based on the cube sphere, the icosahedron, or other divisions of the sphere into patches. For the cube sphere see Sadourny (1972) or Rancic et al. (1996). The icosahedral grid was introduced by Williamson (1968), Sadourny et al. (1968), and Cullen (1974). These early approaches suffered from the wavenumber-5 problem, which was considered to be a severe drawback. The error consists of the false appearance of a wavenumber 5, such as in the wavenumber-4 test case of Williamson et al. (1992). Masuda and Ohnishi (1986) reported the first successful application of the finite-volume method on the icosahedral grid. Steppeler and Prohl (1996) avoided the wavenumber-5 problem by maintaining approximation order of 2 near the patch boundaries. Several research groups have developed icosahedral grids using their own numerical approaches and tested them by mainly using the shallow-water model. For climate research
see Masuda and Ohnishi (1986), Heikes and Randall (1995a,b), and Ringler et al. (2000). A forecast model was developed by Steppeler and Prohl (1996) and Majewski et al. (2002). An icosahedral model for high-resolution simulations on the Earth Simulator was developed by Tomita and Satoh (2004). Refer to Tomita and Satoh (2004) for a more complete discussion of models based on the icosahedron.

The methods discussed above are of approximation order 2, being based either on the finite-volume method or the second-order finite-difference stencil of Baumgardner and Fredrickson (1985). Tomita et al. (2001) use the spring dynamics method to make the grid approximately uniform, so that the finite-volume method becomes more accurate. With this method a grid is obtained by subdividing the original icosahedral grid. This grid is then changed in order to result into more equally sized triangles. While this method makes the finite-volume scheme more accurate, the resulting corners of the triangles do not line up very well near great circles. The Baumgardner second-order difference stencil is difficult to generalize to approximation order 3 (J. R. Baumgardner 2000, personal communication).

In the present paper we use great circle grids. The original patches of the icosahedron are subdivided such that the corners of the rhomboids of discretization are on great circles. This feature is important for the construction of high-order difference stencils having a similar computational efficiency as operators in rectangular grids. Points aligned on great circles are achieved by constructing the grid in a two-step procedure, similar to that used by Williamson (1968). The grid is first constructed on the bilinear surface defined by the corners of a patch and then projected to the sphere. This process is used here to obtain the alignment of the points on great circles. The methods of Sadourny et al. (1968), Baumgardner and Fredrickson (1985), and Heikes and Randall (1995) cannot generate grid points aligned on great circles. The change to great circle grids offers the possibility to achieve a high approximation order rather simply. By taking the derivative along a line of grid points on a great circle, classical difference methods can be applied and it will not be too difficult to transfer local models, such as the Weather Research and Forecasting (WRF) model (Skamarock 2004) or the Consortium for Small-scale Modeling (COSMO) model [formerly known as the Local Model (LM)] (Steppeler et al. 2003), to the sphere and preserve the order of approximation. It will be necessary to write interfaces between the patches and change the difference stencils to allow for a nonorthogonal coordinate. Achieving positive solutions for moisture or chemistry fields will also transfer easily from the local models, as flux correction can be done along great circles. This is not done here because this paper is limited to the shallow-water equations.

The transfer of conservation properties to the sphere for approximation order greater than two is more difficult and this problem is not treated in this paper. Possibilities to include conservation will be discussed in section 6.

The concept mentioned above is demonstrated by a shallow-water model on the sphere. Fourth-order finite differences are used and care is taken that the derivatives remain fourth order even for irregular grid spacing. The time differencing is done by the fourth-order Runge–Kutta method, using explicit time stepping in a similar way as suggested by Wicker and Skamarock (2002) for the third order, the main difference being that fourth-order Runge–Kutta methods can be used without intrinsic or explicit diffusion. For simplicity we do not use substepping, but rather employ time steps small enough that explicit integration of the fast waves is possible. This is a simple way to make the point that high-order methods are feasible. For operational efficiency the time-splitting third-order Runge–Kutta method by Wicker and Skamarock (2002) should be used. The transition from the methods used here to that method would only require some technical development. In particular it is possible to use the fifth-order differentiation in the same way as Wicker and Skamarock (2002).

The interaction between the patches is achieved by continuing a line of grid points beyond patch boundaries and creating boundary values by serendipity interpolation. This method can be considered as a variant of the Skamarock and Klemp (1993) scheme to create adaptive models. Serendipity grids are common with high-order finite-element methods. Steppeler (1976) used them in the context of a finite-difference method. In this paper we aim for a third-order method uniformly in space and time, thus, we limit ourselves to third-order serendipity interpolation.

Section 2 will describe the grid. The serendipity grid, the creation of boundary values beyond patches, and interpolation will be described in section 3. Section 4 will define the difference stencils and section 5 will present some results, including convergence tests using test cases 2 and 6 of Williamson et al. (1992).

2. Grid generation

The grid will be constructed by developing it first on the inscribed bilinear surface and then projecting it to the sphere. The kind of grid used is defined by the
parameter NP, determining the number of patches to be used. Even though this paper is limited to the icosahedron (NP = 5), this construction of the grid will allow us to extend the method to other quasi-homogeneous grids, such as the cube sphere. The angle of $2\pi$ at the Pole is divided into NP parts, defining NP patches with a common corner at the North Pole and the same is done for the South Pole. This results in two NP rhomboidal patches sharing $2NP + 2$ corners between them. The definition of these basic patches and their corners will be described in the next paragraph. The construction will lead to the cube sphere for NP = 3 and to the icosahedron for NP = 5.

The corners of the basic patches and the grid points are represented as three-dimensional vectors:

$$\mathbf{r} = \left( \begin{array}{c} r_1 \\ r_2 \\ r_3 \end{array} \right).$$

The components are the projections of $\mathbf{r}$ into the coordinate directions $x$, $y$, and $z$.

Here we will construct the grid on the unit sphere. The grid on the earth with radius $a$ can then be obtained by multiplying all grid points by $a$. A point $\mathbf{r}$ is on the unit sphere if

$$|\mathbf{r}| = \sqrt{r_1^2 + r_2^2 + r_3^2} = 1.$$

We will, however, also use points outside the sphere as auxiliary points for the construction of the spherical grid. We now define $2NP + 2$ points on the sphere being the corners of the basic rhomboids. The North and South Poles are

$$\mathbf{r}_1 = \left( \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right) \quad \text{and} \quad \mathbf{r}_{2NP+2} = \left( \begin{array}{c} 0 \\ 0 \\ -1 \end{array} \right). \quad (1)$$

Let the side of the rhomboid to be constructed be $s$. Further NP corners in the Northern Hemisphere are defined as

$$\mathbf{r}_{\nu+1} = \left( \begin{array}{c} \cos[2(\nu - 1)\pi/NP] \sin(s) \\ \sin[2(\nu - 1)\pi/NP] \sin(s) \\ \cos(s) \end{array} \right), \quad \nu = 1, \ldots, NP. \quad (2)$$

The corresponding NP corners in the Southern Hemisphere are

$$\mathbf{r}_{\nu+NP+1} = \left( \begin{array}{c} \cos[2(\nu - 1)\pi/NP + \pi/NP] \sin(s) \\ \sin[2(\nu - 1)\pi/NP + \pi/NP] \sin(s) \\ -\cos(s) \end{array} \right), \quad \nu = 1, \ldots, NP. \quad (3)$$

The 2NP + 2 corners defined above are on the unit sphere.

These points are used to define the basic rhomboidal patches and the grid. A rollup of the patches and planes is shown in Fig. 1. The borders of the spherical rhomboidal patches are defined by connecting some of the points by great circles. For the construction of the grid the inscribed patches will also be used. The boundaries of the inscribed bilinear surfaces are obtained by connecting some points of opposing boundaries by straight lines. For NP = 3, the inscribed surfaces are the six surfaces of the cube and the spherical patches are the projections of these surfaces to the sphere. For the cube, the bilinear inscribed surfaces are planes, which means that any straight line connection between two boundary points are in this surface. The distance of two points $\mathbf{a}$ and $\mathbf{b}$ on the inscribed surface is

$$\text{dist}(\mathbf{a}, \mathbf{b}) = |\mathbf{a} - \mathbf{b}|.$$
numbered 1–rilaterals of any shape. The corners of this patch are the geometry of the grid can be constructed for quad-
general shape than a rhomboid is shown to indicate that in Fig. 2, we have patches
points defined above. Corner 1 is point \( r_1 \) for patches 1, \( \ldots \), NP and point \( r_{1+NP+2} \) for patches \( i = NP + 1, \ldots, 2NP - 1 \) and \( r_2 \) for \( i = 2NP \). Corner 2 is point \( r_{i+1} \) for patches \( i = 1, \ldots, 2NP \). Corner 3 is point \( r_{i+NP+1} \) for patches \( i = 1, \ldots, NP \) and \( r_{2NP+2} \) for patches \( i = NP + 1, \ldots, 2NP \). Corner 4 for patch \( i \) is \( r_{i+2} \) for \( i = 1, \ldots, NP - 1 \) and \( i = NP + 1, \ldots, 2NP - 1 \), \( r_2 \) for patch \( i = NP \), and \( r_{2NP+2} \) for patch \( i = 2NP \).

The inscribed bilinear grid is illustrated in Fig. 2, showing one side of the inscribed bilinear polygon, which could be patch \( P_i \) of Fig. 1. In Fig. 2 a more general shape than a rhomboid is shown to indicate that the geometry of the grid can be constructed for quadrilaterals of any shape. The corners of this patch are numbered 1–4 as shown in Fig. 2. The edge belonging to the corners \( i, j \) is the set of all points \( r_i + \lambda (r_j - r_i) \), when \( \lambda \) is a number between 0 and 1. There are two sets of opposite sides for each rhomboid: 1, 2 and 4, 3; and 2, 3 and 1, 4. For any two points \( e_1, e_2 \), the point \( e_1 + \lambda (e_2 - e_1) \) is said to be on the connecting line of these points and \( \lambda \) is defined to be the dividing factor. The bilinear surface corresponding to a rhomboid is defined as the set of connecting lines of points of equal dividing factor on opposing sides. It can easily be verified that this definition does not depend on the set of opposing sides chosen. In fact, the following geometric property, being illustrated in Fig. 2 holds on bilinear surfaces, being true even when the four points of the rhomboid do not lie on a plane.

Choose a point of dividing factor \( \lambda_1 \) on each of the opposing sides 1, 2 and 4, 3 and call the connecting line of these points \( l_1 \), whose end points are numbered 5 and 7 in Fig. 2. Choose another set of points of dividing factor \( \lambda_2 \) on opposing sides 2, 3 and 1, 4 and call the connecting line \( l_2 \). The end points of this line are indicated by 8 and 6 in Fig. 2. Then \( l_1 \) and \( l_2 \) intersect in a point \( P \) and \( P \) divides line \( l_2 \) by the dividing factor \( \lambda_1 \) and line \( l_1 \) by \( \lambda_2 \). Using the indices of points indicated in Fig. 2, we have

\[
\lambda_1 = \frac{\text{dist}(5, 1)}{\text{dist}(2, 1)} = \frac{\text{dist}(7, 4)}{\text{dist}(3, 4)} = \frac{\text{dist}(P, 8)}{\text{dist}(6, 8)}
\]

and

\[
\lambda_2 = \frac{\text{dist}(6, 2)}{\text{dist}(3, 2)} = \frac{\text{dist}(8, 1)}{\text{dist}(4, 1)} = \frac{\text{dist}(P, 5)}{\text{dist}(7, 5)}.
\]

This theorem can easily be used to construct regular grids on the inscribed bilinear surfaces. The grid points \( r_{i,j} \) are defined on the bilinear grid as follows:

![Fig. 2. The definition of the inscribed bilinear grid. The grid will be defined for each of the rhomboidal patches \( p_1 \) to \( p_{10} \) shown in Fig. 1, but the construction is valid for all quadrilaterals. The corners indicated by \( e_1, e_2, e_3, \) and \( e_4 \) are the corners being chosen from the \( r_1, \ldots, r_{2NP} \) in Fig. 1. (bottom) The corners are numbered 1–4 and can be arbitrary vectors in 3D space. The lines \( l_1 \) and \( l_2 \) divide the two sides they intersect by equal ratios defined to be the dividing factors \( \lambda_1 \) and \( \lambda_2 \). They always intersect in a point \( P \), even when the four corners are not in one plane. Here \( l_1 \) and \( l_2 \) divide each other by the same dividing factors \( \lambda_1 \) and \( \lambda_2 \).

\( \lambda_1 = \frac{\text{dist}(5, 1)}{\text{dist}(2, 1)} = \frac{\text{dist}(7, 4)}{\text{dist}(3, 4)} = \frac{\text{dist}(P, 8)}{\text{dist}(6, 8)} \)

and

\( \lambda_2 = \frac{\text{dist}(6, 2)}{\text{dist}(3, 2)} = \frac{\text{dist}(8, 1)}{\text{dist}(4, 1)} = \frac{\text{dist}(P, 5)}{\text{dist}(7, 5)} \).

\( r_{i,j} \) are defined on the bilinear grid as follows:
For example, of grid lengths between 1000 and 120 km the ratio mentioned above changed only by about 1% for the grid in Heikes and Randall (1995). The same is true for our grid and the ratio reported above is similar to that in Heikes and Randall (1995) and is about the same as for the grid in Baumgardner and Fredericson (1985).

3. Serendipity interpolation

To compute the grid point values on the patch boundaries, it is necessary to pose boundary values. Such boundary values to be interpolated are all dynamic fields. For the case of the shallow-water equations these are the velocity components and the height field. The grid is extended beyond the boundaries as shown in Fig. 4 and these values are posed by interpolation within the neighboring patch. All kinds of interpolation methods could be used for this purpose, such as cascading interpolation as used in semi-Lagrangian schemes. Here serendipity interpolation (see Brenner and Scott 2002) is used which is marginally cheaper. It uses only 12 grid points from one and the same patch. In comparison cascading interpolation uses 16 points.

Define the points inside the set of \( 4^p \) grid points used for the interpolation of a serendipity patch. Than the interpolation has the property that a field at a point inside such a patch is interpolated using only grid points from the same patch.

The method used here to connect patches with different grid orientations was used by Skamarock and Klemp (1993) to refine the resolution. For this application the serendipity interpolation has the advantage that grid overlaps are not necessary. The smallest refined area should then be a serendipity patch of \( 4 \times 4 \) points and rather irregular areas composed of such patches could be admitted.
P1 and P2 in Fig. 1. The patch for the grid points to be time of grid points common to both patches, such as between patches thick solid line is the boundary between patches containing a line with 4 part of the grid in Fig. 3 bordering the north pole: Let us call \( r \) the grid point to be interpolated. Then this point is within a \( 4 \times 4 \) set of points in a neighboring serendipity patch. Let us number these points as 0, 1, 2, and 3 in both directions, as shown in Fig. 4. The thick vertical line in Fig. 4 could be the edge between patches P1 and P2 in Fig. 1. Then each grid point \( r_{ij} \) in a serendipity patch is defined by two indices \( i \) and \( j \), both running from 0 to 3. The inner points 1, 1; 1, 2; 2, 1; and 2, 2 will not be used for the interpolation. The interpolation and all other computations are done on the sphere, the bilinear grid being used only for the construction of the grid.

Two lines of points are needed outside the boundary line. Here \( \nu_{\text{max}} \) and \( \mu_{\text{max}} \) must be multiples of 3. The points with \( \nu = \nu_{\text{max}} \) or \( \mu = \mu_{\text{max}} \) are identical to points with \( \nu = 0 \) or \( \mu = 0 \) in neighboring patches. Care must be taken to define such points in a unique way, otherwise roundoff errors may accumulate and produce instabilities in long integrations.

Let us call \( r \) the grid point to be interpolated. Then this point is within a \( 4 \times 4 \) set of points in a neighboring serendipity patch. Let us number these points as 0, 1, 2, and 3 in both directions, as shown in Fig. 4. The thick vertical line in Fig. 4 could be the edge between patches P1 and P2 in Fig. 1. Then each grid point \( r_{ij} \) in a serendipity patch is defined by two indices \( i \) and \( j \), both running from 0 to 3. The inner points 1, 1; 1, 2; 2, 1; and 2, 2 will not be used for the interpolation. The interpolation and all other computations are done on the sphere, the bilinear grid being used only for the construction of the grid.

The angular distance arc of two grid points in a serendipity patch is defined as

\[
\text{arc} = \arccos(r_{ij+1,i} \cdot r_{ij}).
\]

Let \( s_1, s_2, s_3, \) and \( s_4 \) be the natural coordinates of \( r \) on the four sides, defined by \( s_1 = s_3 \) and \( s_2 = s_4 \). These natural coordinates take values between 0 and 1. Using field values at the points \( r_{0,0}; r_{0,3}; r_{3,0}; \) and \( r_{3,3} \) the field \( \Phi \) is bilinearly interpolated to the eight points \( r_{0,1}; r_{0,2}; r_{3,1}; r_{3,2}; r_{1,0}; r_{1,3}; r_{2,1}; \) and \( r_{2,3} \).

The bilinearly interpolated value to the point \( r \) is denoted as \( \Phi \). The difference of \( \Phi \) to the linearly interpolated values on each of the four sides is interpolated in third order to the coordinates \( s_1, s_2, s_3, \) and \( s_4 \), resulting into values

\[
\delta \Phi_1, \delta \Phi_2, \delta \Phi_3, \text{ and } \delta \Phi_4.
\]

Referring to \( \delta \Phi, \delta \Phi' \) as the results of one-dimensional linear interpolations of \( \delta \Phi \) between the sides 1 and 3, and 2 and 4, respectively, then the result of the serendipity interpolation is

\[
\Phi_{\text{ser}} = \Phi + \delta \Phi + \delta \Phi'.
\] (8)

4. Difference stencils and definition of the shallow-water discretization

Following Baumgardner and Frederickson (1985), a different polar coordinate system is used in every grid point. The coordinate system is defined by vectors \( n_x, n_y, \) and \( n_z \). For input and output purposes the coordinate system is used where \( n_z \) is the direction of the North Pole:

Fig. 5. Test case 1 of Williamson. Advection of a rotational structure in a homogeneous velocity field around the equator. (a) The solution after a forecast time of 12 days, when a full rotation is achieved. The solution is shown as isolines having a distance of 100 m. The error is indicated by shading. (b) As in (a), but for the third-order upstream scheme. (c) As in (b), but for the positive definite scheme.
The velocity for input output purposes is defined by its components belonging to the polar coordinate system defined by \( \mathbf{n}_x, \mathbf{n}_y, \mathbf{n}_z \). For a grid point \( \mathbf{e}_{\nu,\mu} \), the vectors defining the local associated polar coordinate are

\[
\mathbf{n}_x^{\nu,\mu} = \left[ \mathbf{e}_{\nu+1,\mu} - (\mathbf{e}_{\nu+1,\mu} \cdot \mathbf{e}_{\nu,\mu}) \mathbf{e}_{\nu,\mu} \right] / \left[ \mathbf{e}_{\nu+1,\mu} - (\mathbf{e}_{\nu+1,\mu} \cdot \mathbf{e}_{\nu,\mu}) \mathbf{e}_{\nu,\mu} \right],
\]

\[
\mathbf{n}_y^{\nu,\mu} = \mathbf{e}_{\nu,\mu}, \quad \text{and}
\]

\[
\mathbf{n}_z^{\nu,\mu} = \mathbf{n}_z \times \mathbf{n}_x,
\]

where \( \cdot \) indicates the scalar product.

Using these vectors, the representation of a vector \( \mathbf{e} \) on the unit sphere in polar coordinates \( \lambda, \varphi \) is

\[
\begin{pmatrix}
\cos(\lambda) \cos(\varphi) \\
\sin(\lambda) \cos(\varphi) \\
\sin(\varphi)
\end{pmatrix}
= e_x^{\nu,\mu} n_x^{\nu,\mu} + e_y^{\nu,\mu} n_y^{\nu,\mu} + e_z^{\nu,\mu} n_z^{\nu,\mu}.
\]

The equation can in particular be used to solve for \( \lambda \) and \( \varphi \) and thus compute local polar coordinates for any grid point on the unit sphere. The velocity in the local coordinate system is given by its components in the directions belonging to the local polar coordinate system being illustrated in Fig. 3 for the coordinate system belonging to one of the grid points.

The shallow-water equations in the coordinate system associated with a grid point are

\[
\begin{align*}
\frac{\partial u}{\partial t} &= -\left( \frac{u}{v} \right) \cdot \nabla u + \left( f + \frac{u}{a} \tan \varphi \right) v - \frac{g}{a \cos \varphi} \frac{\partial h}{\partial \lambda} - d_1 \Delta^2 u, \\
\frac{\partial v}{\partial t} &= -\left( \frac{u}{v} \right) \cdot \nabla v - \left( f + \frac{u}{a} \tan \varphi \right) u - \frac{g}{a \cos \varphi} \frac{\partial h}{\partial \lambda} - d_2 \Delta^2 v, \\
\frac{\partial h}{\partial t} &= -\left( \frac{1}{v} \right) \cdot \nabla h - \frac{h}{a \cos \varphi} \left( \frac{\partial u}{\partial \lambda} + \frac{\partial v}{\partial \varphi} \cos \phi \right) - d_3 \Delta^2 h,
\end{align*}
\]

where \( a \) is the earth radius, \( f \) is the Coriolis parameter, and \( \Delta \) is the Laplacian operator. Here \( d_1, d_2, \) and \( d_3 \) are the numerical diffusion coefficients. In all experiments reported in this paper there will be no diffusion of the \( h \) field, and \( d_3 = 0 \).

As every grid point uses another polar coordinate system, these dynamic equations are applied to compute the time derivative of just the one grid point defining the coordinate. Therefore \( \lambda \) and \( \varphi \) are only used for values near 0.

The discrete equations are derived from Eq. (12) by replacing the derivatives by approximated derivatives to be defined in the following. This simple procedure depends on using the Arakawa A grid, which is used here, similar as in other icosahedral approaches, such as Baumgardner and Frederickson (1985). More complicated staggered grid representations could be used in the same way. A limited-area model, such as WRF, could be transferred to the icosahedral grid with little effort. The approximation order of the local model would then be transferred to the globe, but not the conservation properties, which the local model might have.

It is possible to use the conservation form of the equations and use staggered forms of the equations. This will, however, not automatically lead to conservation. Prospects to design conserving high-order schemes on the icosahedron will be discussed in section 6.

The coordinate lines for the indices \( \nu \) and \( \mu \) are illustrated in Fig. 2. We use a 5-point stencil to define a fourth-order differentiation along the coordinate lines \( \nu \) and \( \mu \). For a field \( \psi \) we number the points by an index \( \eta \) taking the values \(-2, -1, 0, 1, 2\). The local indices to define the stencil are illustrated in Fig. 2. The differentiation is done on the sphere, not on the inscribed surface. With each index \( \eta \) a grid point \( \mathbf{e}_\eta \) is associated. With each of these points we associate \( x_\eta \) values, the grid distances to the point with index \(-2\). These are used to obtain the grid distances for the numerical differentiation. For the example of the derivative in differentiation in \( \nu \) direction these are

\[
\begin{align*}
x_\eta &= \arccos (\mathbf{e}_{\nu+1} \cdot \mathbf{e}_{\nu-2}) & \text{for} \quad \eta = -2, \quad -1, 0, 1, 2.
\end{align*}
\]

The fourth-order approximation of the derivative of a field \( \psi \) in \( \nu \) direction is

\[
\psi_x = a_{-2} \psi_{-2} + a_{-1} \psi_{-1} + a_0 \psi_0 + a_1 \psi_1 + a_2 \psi_2, \quad \text{with}
\]

\[
a_{\eta} = \frac{\partial g^\eta(x_\eta)}{\partial x},
\]

\[\text{(14)}\]
where \( g^{(n)}(x) \) is defined as the fourth-order polynomial in \( x \) which is 1 at \( x = 0 \) and 0 at all other values of \( x \). For the differentiation of velocities all components have to be transformed to the coordinate system at point \( \eta = 0 \).

Here \( \psi_x \), the fourth-order approximation of \( \psi \) in direction \( x \) is defined in a similar way. The derivatives \( \psi_x \), and \( \psi_y \) are obtained as follows:

\[
\psi_x = \psi_x \quad \text{and} \quad \psi_y = (\psi_y / \sin \alpha) - c t g \alpha \psi_x, \tag{15}
\]

where \( \alpha \) is the angle between the coordinate lines \( \nu \) and \( \mu \).

The Laplacian operator is discretized as in Baumgardner and Fredrickson (1985) and Majewski et al. (2002). The other terms are discretized using the fourth-order derivative approximation defined above. With the spatial approximation achieved, the time integration is done by the fourth-order Runge–Kutta method. We write the equations in Eq. (12) in the following compact way:

\[
\frac{\partial}{\partial t} \begin{pmatrix} u \\ v \\ h \end{pmatrix} = \frac{\partial}{\partial t} \begin{pmatrix} \text{rs}_1(\text{uh}) \\ \text{rs}_2(\text{uh}) \\ \text{rs}_3(\text{uh}) \end{pmatrix} = \text{rs}(\text{uh}). \tag{16}
\]

Then the fourth-order Runge–Kutta time stepping for step \( dt \) is defined as follows:

\[
k_1 = d\text{rs}(\text{uh}), \quad k_2 = d\text{rs}(\text{uh} + k_1/2), \quad k_3 = d\text{rs}(\text{uh} + k_2/2), \quad k_4 = d\text{rs}(\text{uh} + k_3), \quad \text{and} \quad \text{uh}_{n+1} = \text{uh}_n + k_1/6 + k_2/3 + k_3/3 + k_4/6. \tag{17}
\]

For a research application the fourth-order Runge–Kutta method offers the advantage that the numerical diffusion can be explicitly defined and its effect investigated. The third-order Runge–Kutta method, defined by Wicker and Skamarock (2002) in combination with substepping of the fast waves is more efficient and probably preferable if operational forecasting is the aim. It uses intrinsic diffusion. For the research application we have in mind here, it is preferable to be able to have control over the diffusion term, in order to investigate its effect on the solution. Some applications, to be reported on in the next section, run for extended forecast times without applying explicit diffusion.

The stability limit for problems with imaginary spatial eigenvalues is given in Durran (1999). The Courant–Friedrichs–Lewy (CFL) stability limit is 2.8. Beyond this limit the amplification factor increases rather strongly, as is typical for high-order schemes. For CFL = 3 the amplification factor is already 1.5. For 2D problems a more strict limit applies, with an additional reduction of the stability limit for the nonorthogonal grid. With \( \alpha \) being the angle between the lines of grid points, the stability limit is CFL = \( 2.8 \times \sin(\alpha/2) \). For a rectangular grid we have \( \sin(\alpha/2) = \sin(45^\circ) = 2^{-1/2} \) and for the icosahedron we have \( \alpha = 72^\circ \). The rather good CFL number makes the fourth-order Runge–Kutta scheme seem efficient, which it is, if a damping of the waves with the highest Courant numbers is acceptable for the application. If, as is the case in this paper, a practically undamping scheme is wanted, the CFL should be limited to 1.5 (Durran 1999).

5. Results

All tests are done for NP = 5, the icosahedron. Test case 1 of Williamson et al. (1992) is the advection of a bell-shaped passive scalar using a rotationally symmetric zonal velocity field. After 12 days the initial distribution of the scalar field is reproduced. Figure 5a shows the predicted field after 12 days and the error field. The error shows some positive and negative values along the path of the advected structure as is typical for fourth-order advection schemes. The error reaches a maximum amplitude of about 1% and is mainly concentrated near the solution and related to the structure of the advected field, not so much to the grid structure. Figure 5b gives the same result for the third-order upstream scheme. The results of a positive definite version of this scheme are shown in Fig. 5c. The scheme uses flux correction among the seven or six points occurring in the diffusion stencil. The well-known sophistications of flux correction schemes are not investigated here, but the results indicate that standard methods to achieve positivity can be employed on great circle grids.

Test case 2 of Williamson et al. (1992) is designed to show an impact of the grid structure on the spatial distribution of the error. It is a stationary rotational flow around the globe. We use here the example of a zonal flow. A time step of 45 s is used. The field and the error of a 5-day forecast for \( v_{\text{max}} = 96 \) is shown in Fig. 6a. No diffusion is applied. The error remains rather small, with a maximum amplitude below 0.3 m. When using diffusion (Fig. 6b), the amplitude of the error is less than 0.0009 m. The diffusion coefficients are \( d_1 = d_2 = k = 1.12 \times 10^{17} \text{ m}^2 \text{ s}^{-1} \) and \( d_3 = 0 \) (for different resolutions the coefficients are scaled as in Chien et al. 1995). These error levels are rather low and are easily exceeded when the meteorological field has a structure to define the error field. For this reason, plots of the error fields will normally not show an impact of the grid.
Fig. 7. Convergence diagram for the homogeneous stationary flow. The $L^2$ and $L^\infty$ norms are shown as a function of time and resolution. (a) Time development of (top) $L^2$ and (bottom) $L^\infty$ norms. (b) As in (a), but with diffusion. (c) The $L^2$ and $L^\infty$ norms as function of resolution for 10-day forecasts with diffusion. Third-order convergence means a curve parallel to the solid line.
structure when solutions involving some curvature of the field are used. Figure 6c shows the forecasted $h$ field and the corresponding error field after a 30-day forecast using $v_{\text{max}} = 96$ with diffusion. The error also remains rather low for this longer integration. Figure 7a (without diffusion) and Fig. 7b (with diffusion) show the time development of the $L_2$ and $L_\infty$ norms (Williamson et al. 1992) as functions of time for different resolutions. With diffusion the error increases in time, with the error level staying below that reported by Tomita et al. (2001). The error growth is monotonic and does not increase and decrease, as was reported by Tomita et al. (2001). Without diffusion the solution will eventually become unstable. With diffusion the error is more constant and rather long integrations are possible. A convergence diagram for test case 2 (with diffusion) is shown in Fig. 7c for 10-day forecasts, showing the $L_2$ and $L_\infty$ norms as a function of resolution ($v_{\text{max}} = \mu_{\text{max}} = 12, 24, 48, 96, \text{ and } 120$). It indicates third-order convergence, as is to be expected from the design of the model.

Test case 6 of Williamson et al. (1992) is used to
demonstrate the third-order convergence for a Rossby–Haurwitz wave of wavenumber 4. Figure 8 shows the forecasted divergence field after 2.5 h. The time step is 45 s. The grid parameters for this solution are $\nu_{\text{max}} = \mu_{\text{max}} = 96$. There is no diffusion used with this solution. The grid structure does not have a noticeable impact on the solution and the field shows smooth behavior near the boundary of the computational patches.

Figure 9 shows the predicted $h$ field after 5 days, using the diffusion coefficients $d_1 = d_2 = k = 1.12 \times 10^{17}$ m$^4$ and $d_3 = 0$. This is approximately the same as used in Majewski et al. (2002). For different resolutions the coefficient is scaled in a similar way as in the model used for reference. According to Thuburn and Li (2000) this solution is unstable in the sense that it is chaotic for forecast times of about 10 days. Therefore for long integration times no convergence of solutions can be expected. Convergence can be investigated for the shorter forecast ranges only. The error distribution is related to the structure of the solution and not to that of the grid. To help the comparison with the results obtained by other authors, it is noted that $\nu_{\text{max}} = \mu_{\text{max}} = 96$ is between levels 5 and 6 in the notation of Heikes and Randall (1995a,b) and levels 6 and 7 in the notation of Tomita et al. (2001).

The error of the solution was investigated using a reference produced by a variant of the National Center for Atmospheric Research (NCAR) spectral transform shallow-water model (STSWM; Chien et al. 1995) with T511 and $dt = 90$ s. The error of the solution is indicated by shading in Fig. 9. Starting with the solution $\nu_{\text{max}} = \mu_{\text{max}} = 12$ the resolution is stepwise doubled, up to a value of $\nu_{\text{max}} = \mu_{\text{max}} = 96$. Figure 10a shows the $L_2$ and $L_\infty$ norms for the height field depending on resolution ($\nu_{\text{max}} = \mu_{\text{max}} = 12, 24, 48, 96, \text{ and } 120$) at day 5 are shown in Fig. 10b. Here $\nu_{\text{max}} = \mu_{\text{max}} = 120$ is similar to level 7 in the notation of Tomita et al. (2001). The $L_2$ and $L_\infty$ norms for the height field for $\nu_{\text{max}} = \mu_{\text{max}} = 120$ at day 5 are below the values reported by Tomita et al. (2001).
Fig. 11. The forecasted $h$ field (m) for forecast times 1, 7, and 14 days. The spacing of the isolines is 200 m.
convergence is confirmed, with a degradation for the higher resolutions. This degradation was not observed in test case 2 (Fig. 7c). It should be noted that the third-order character of the solution comes from the third-order boundary interpolation scheme only. Space and time differencing are fourth order.

Using visual comparison the solution after longer times of integration is rather similar to that obtained by other models and the structure of the solution for test case 6 is maintained. Figure 11 shows the time evolution of the solution. There is a coherent development and a maintenance of the wavenumber-4 structure up to 14 days. The 14-day integrations is comparable to Fig. 20 of Tomita et al. (2001).

6. Conclusions and outlook

The construction of third-order finite-difference schemes on grids of icosahedral type is possible using classical difference stencils on great circles. This offers the possibility to transfer time integration methods known from limited-area models to the sphere. The test solution does not involve a larger error at places where the grid orientation changes, such as at boundaries of patches. The third-order approximation was verified by a convergence test.

A new feature of the grid is the alignment of grid points on great circles. This feature was applied here to achieve third-order schemes rather easily, but other methods used with icosahedral grids could be applied as well. Because of the design of the grid aligned on great circles, limited-area models (e.g., WRF) could be transferred to the sphere with little changes to the program. The order of approximation of the local model would transfer to the icosahedron if it is possible to reprogram it in a nonorthogonal coordinate scheme. This means, that the third- or higher-order approximation of WRF would immediately be available after transition to the icosahedral grid. With conservation properties the transfer to the icosahedron is more difficult, as this is often done using the concept of the dual grid, which for plane rectangular grids would result again in a rectangular grid. For the icosahedron the dual grid involves hexagonal grid cells. A staggered grid on the icosahedron involving hexagonal cells was proposed by Ringler and Randall (2002) and could be applied in the same way in the grid constructed here. This approach, however, is limited to approximation order 2. A different approach to conservation proposed by Steppeler (1986) is based on the finite-element method and achieves conservation on the original grid, without using the dual grid. This approach allows approximation orders higher than 2. It should be noted, that there is the prospect to use an existing model to compute the Galerkin integrals, which could be corrected for conservation. Therefore, this could again be done by a small change in the model. For example, because WRF is a third-order model in space and time, there would be the prospect of writing a conservation module for it in order to achieve a third-order model conserving mass and energy.

Acknowledgments. This work resulted from a cooperation between DWD and NCAR. Part of the work was done while the first author visited NCAR with travel support from NCAR. The authors thank the anonymous reviewers for their constructive comments.

REFERENCES


Sadourny, R., 1972: Conservative finite-difference approxima-
——, A. Arakawa, and Y. Mintz, 1968: Integration of the nondi- verge
——, and J. Klemp, 1993: Adaptive grid refinements for two-di-men
dimensional and three-dimensional nonhydrostatic atmosphe
Steppeler, J., 1976: The application of the second and third degree