Implementation of an Atmosphere–Ocean General Circulation Model on the Expanded Spherical Cube

ALISTAIR ADCROFT, JEAN-MICHEL CAMPIN, CHRIS HILL, AND JOHN MARSHALL

Massachusetts Institute of Technology, Cambridge, Massachusetts

(Manuscript received 25 October 2002, in final form 16 January 2004)

ABSTRACT

A hydrodynamical kernel that drives both an atmospheric and oceanic general circulation model is implemented in general orthogonal curvilinear coordinates using the finite-volume method on the sphere. The finite-volume method naturally describes arbitrary grids, and use of the vector-invariant form of the momentum equations simplifies the generalization to arbitrary coordinates. Grids based on the expanded spherical cube of Rancic et al., which contain eight singular points, are used. At these singularities the grid is nonorthogonal. The combined use of vector-invariant equations and the finite-volume method is shown to avoid degeneracy at these singular points.

The model is tested using experiments proposed by Williamson et al. and Held and Saurez. The atmospheric solutions are examined seeking evidence of the underlying grid in solutions and eddy statistics. A global ocean simulation is also conducted using the same code. The solutions prove to be accurate and free of artifacts arising from the cubic grid.

1. Introduction

The most commonly used coordinate system in oceanic and atmospheric general circulation models is the latitude–longitude grid characterized by two coordinate singularities—one at each geographic pole. The singularities themselves are not necessarily problematic, but associated with them is a convergence of meridians that leads to ever-decreasing zonal grid spacing as one approaches the poles. These small grid sizes impose severe limits on the time step demanded by explicit time-stepping schemes. Approaches to overcoming this limitation include the use of implicit (unconditionally stable) time stepping for the fast modes, semi-Lagrangian methods for large Courant numbers, and filters that stabilize the model against computational instabilities (truncation or scaling of Fourier modes in the zonal direction). In the ocean, most of these techniques are not easily applied—semi-Lagrangian methods are difficult to make absolutely conservative and have problems in regions where trajectories intersect solid boundaries, and the use of zonal filters is likewise difficult in the presence of irregular solid boundaries. A more elegant solution is to avoid the converging meridian problem entirely by using a different coordinate system or gridding of the sphere.

Many coordinate systems have been investigated for use in global modeling. These include the gnomonic cubic grid of Sadourny (1972) and Ronchi et al. (1996), and icosahedral grids of Ringler and Randall (2002). Most general circulation models, whether oceanic or atmospheric, are written using orthogonal curvilinear coordinates. Ours, in particular, the Massachusetts Institute of Technology (MIT) general circulation model (GCM; Marshall et al. 1997a,b), uses the finite-volume paradigm (Adcroft et al. 1997) for discretization, which is inherently able to describe curvilinear grids, but, the model equations assume orthogonal coordinates. Implementing the model in a general curvilinear coordinate system would require significant algorithmic changes (these issues are summarized in the appendix). Further, the model is written assuming quadrilaterally shaped finite volumes, so adapting the code to work with some other basic shape would be difficult. We, therefore, restrict ourselves to grids that use *orthogonal* curvilinear coordinates. Such considerations preclude grids constructed from triangles or hexagons (Ringler and Randall 2002) and many of the quadrilaterally based grids proposed to date, such as the nonorthogonal gnomonic cubic grid (Sadourny 1972; Ronchi et al. 1996; Purser and Rancic 1998). However, one class of grids, described by Rancic et al. (1996) and McGregor (1996), uses conformal mapping and are, therefore, orthogonal. The conformally expanded spherical cube has near-uniform coverage of the sphere and, thus, allows for a longer explicit time step. However, the expanded cubic grids are only orthogonal in the interior of the coordinate

Corresponding author address: Alistair Adcroft, 77 Massachusetts Ave., Cambridge, MA 02139.
E-mail: adcroft@mit.edu

system; they are not orthogonal at coordinate singularities (corners of the cube). One issue that we address here is whether an orthogonal coordinate model can be implemented on such grids without incurring significant errors in the vicinity of nonorthogonal singular points.

In this paper, we will outline the issues we encountered when implementing a hydrodynamical kernel on a conformally mapped expanded cube (spherical cubic grid). The kernel can be applied to either the ocean or the atmosphere by exploiting an isomorphism between the equations that govern the two fluids, as described in Marshall et al. (2004, manuscript submitted to *Mon. Wea. Rev.*). Because we will be concerned here with asymmetries in the solutions due to anisotropic properties of the grid, we concentrate on idealized atmospheric solutions (Williamson et al. 1992; Held and Saurez 1994), but also present some oceanographic solutions. In section 2 we discuss the form of the equations and our general discretization. In section 3 we describe the specific discretization adopted and discuss our treatment of the coordinate singularities. In section 4 we present modified orthogonal grids that build on the original conformally mapped expanded spherical cube. Numerical results for two- and three-dimensional calculations, both atmospheric and oceanic, are presented in section 5.

2. Primitive equations in curvilinear coordinates

The equations of motion in curvilinear orthogonal coordinates (λ, ϕ) can be written in two distinct ways: (i) the conservative form, for which the horizontal momentum equations are

$$\partial_t u + u\partial_x u + v\partial_y u + \omega\partial_p u - \left(f + \frac{v\partial_x h_\phi}{h_\phi} - \frac{u\partial_y h_\lambda}{h_\lambda}\right)v + \partial_x \Phi = \mathcal{F}_u \quad (1a)$$

$$\partial_t v + u\partial_x v + v\partial_y v + \omega\partial_p v + \left(f + \frac{v\partial_x h_\phi}{h_\phi} - \frac{u\partial_y h_\lambda}{h_\lambda}\right)u + \partial_y \Phi = \mathcal{F}_v, \quad (1b)$$

where u, v are the wind speeds in the λ and ϕ directions, ω is the rate of change of pressure (vertical coordinate), Φ is the geopotential and $f = f(\lambda, \phi)$ is the Coriolis parameter, $\mathcal{F}_u$ and $\mathcal{F}_v$ represent sources and sinks of momentum, and x and y are curvilinear distances along coordinates λ and ϕ and will be defined below; and (ii) the vector invariant form (as used by Sadourny 1975), which are

$$\partial_t u - (f + \zeta)v + \omega\partial_p u + \partial_x(\Phi + K) = \mathcal{F}_u \quad (2a)$$

$$\partial_t v + (f + \zeta)u + \omega\partial_p v + \partial_y(\Phi + K) = \mathcal{F}_v. \quad (2b)$$

Here, the relative vorticity is given by

$$\zeta = \frac{1}{h_\phi}\partial_x(h_\phi v) - \frac{1}{h_\lambda}\partial_y(h_\lambda u) \quad \text{and}$$

$$K = \frac{1}{2}(u^2 + v^2).$$

In both forms the map factors, h_λ and h_ϕ describe the distortion of the grid and relate the curvilinear distances $(\Delta x, \Delta y)$ to the coordinate changes $(\Delta\lambda, \Delta\phi)$:

$$\Delta x = h_\lambda \Delta\lambda \quad \text{and} \quad \Delta y = h_\phi \Delta\phi.$$

As described in the appendix the map factors are related to the metric tensor of the coordinate transformation g_{ij} by

$$h_\lambda = \sqrt{g_{11}} \quad \text{and} \quad h_\phi = \sqrt{g_{22}}.$$

For orthogonal coordinates the metric tensor is diagonal, $g_{12} = g_{21} = 0$, so that orthogonal coordinates can be seen as a simplifying subset of general curvilinear coordinates.

In the above equations we have used spatial derivatives ∂_x and ∂_y written in terms of physical distance rather than with derivatives in the curvilinear coordinate ∂_λ and ∂_θ. Switching from one to another involves a simple transformation,

$$\partial_x \theta = \frac{1}{h_\lambda}\partial_\lambda \theta \quad \text{and} \quad \partial_y \theta = \frac{1}{h_\phi}\partial_\phi \theta.$$

Use of ∂_x and ∂_y renders the equations compact and understandable by avoiding the introduction of coefficients in front of all of the derivatives. However, the principle reason for their use is that the equations are then naturally discretized using the finite-volume method. Map factors that would otherwise be explicit in the equations are incorporated into the grid of volumes, areas, and lengths that describe a finite-volume grid. As we now describe, this approach leads to a different result than if we had discretized the equations in curvilinear coordinates.

For example, consider discretizing the divergence of a horizontal flux $\nabla \cdot \mathbf{F}$. Expressed in the curvilinear coordinates (λ, ϕ), this is

$$\nabla \cdot \mathbf{F} = \frac{1}{h_\lambda h_\phi}[\partial_\lambda(h_\phi F_\lambda) + \partial_\phi(h_\lambda F_\phi)],$$

which can be discretized using finite differences as

$$\frac{1}{h_\lambda h_\phi}[\partial_\lambda(h_\phi F_\lambda) + \partial_\phi(h_\lambda F_\phi)] \rightarrow \frac{1}{h_\lambda h_\phi}\left[\frac{1}{\Delta\lambda}\delta_i(h_\phi F_\lambda) + \frac{1}{\Delta\phi}\delta_j(h_\lambda F_\phi)\right], \quad (3)$$

where $\delta_i \theta \equiv \theta_{i+1/2} - \theta_{i-1/2}$ and $\delta_j \theta \equiv \theta_{j+1/2} - \theta_{j-1/2}$. In physical (x, y) coordinates, instead, it is natural to write (using Gauss' divergence theorem)

$$\nabla \cdot \mathbf{F} \rightarrow \frac{1}{\mathcal{A}} \iint \nabla \cdot \mathbf{F}\, d\mathcal{A} = \frac{1}{\mathcal{A}} \oint \mathbf{F} \cdot dl$$
$$= \frac{1}{\mathcal{A}} [\delta_i(\Delta y F_\lambda) + \delta_j(\Delta x F_\phi)], \quad (4)$$

where

$$\mathcal{A} = \iint dx\, dy. \quad (5)$$

If the integral in Eq. (4) is expressed in terms of the curvilinear coordinates (λ, ϕ), the result is unchanged as long as the area $\mathcal{A}$ is unchanged. The finite-difference approximation in orthogonal curvilinear coordinates (3) is recovered if the true area in Eq. (4) is replaced with the approximation

$$\mathcal{A} \approx h_\lambda h_\phi \Delta\lambda \Delta\phi. \quad (6)$$

This subtle difference between finite-volume and finite-difference interpretations leads to significant differences in regions of high curvature with finite resolution. One advantage in using (6) is that formal conservation of higher moments is easier to achieve. We have not tried to find a higher-moment-conserving discretization using (5), but see no reason why one should not exist.

The vector-invariant form of the momentum equations (2) are simpler to implementation than the conservative form (1); the metric terms and map factors are ''hidden'' inside the definition of vorticity and definitions of grid lengths and areas and, thus, do not need to be discretized directly in the momentum equations. Moreover, we will show later that the vector-invariant equations, when discretized using the finite-volume method, have a distinct advantage over (1) when the coordinate system has singularities.

In summary, the three-dimensional equations that must be discretized in the hydrostatic version of our hydrodynamical kernel are

$$\partial_t u - (f + \zeta)v + \omega\partial_p u + \partial_x(K + \Phi) = \mathcal{F}_u, \quad (7a)$$
$$\partial_t v + (f + \zeta)u + \omega\partial_p v + \partial_y(K + \Phi) = \mathcal{F}_v, \quad (7b)$$
$$\Pi_p \theta + \partial_p \Phi = 0, \quad (7c)$$
$$\partial_x u + \partial_y v + \partial_p \omega = 0, \quad (7d)$$
$$\partial_t \theta + \partial_x(\theta u) + \partial_y(\theta v) + \partial_p(\theta\omega) = Q/\Pi, \quad (7e)$$

where Q is the sources and sinks of heat, $\Pi = c_p(p/p_c)^\kappa$ is the Exner function, and θ is potential temperature.

3. Implementation of the hydrodynamical kernel on the conformally expanded cube

Implementation of the hydrodynamical kernel will be described in two parts: (i) computational issues involving the mapping of the cubic topology onto the rectangular or linear memory of a parallel computer, and (ii) discretization of the hydrodynamical equations on the expanded cubic grid.

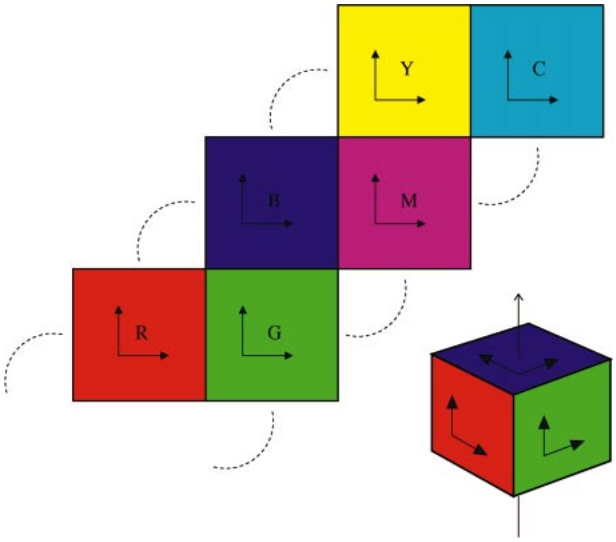

FIG. 1. The logical arrangement of tiles used in implementing the cubic topology. Note that the coordinate systems match across the interior edges, while the exterior edges (dashed arcs) need either a $+\pi/2$ or $-\pi/2$ rotation of coordinates.

a. Computational implementation

The grid proposed by Rancic et al. (1996) uses a conformal mapping of a face of a cube onto a tile on the surface of a sphere. The resulting coordinate system on the sphere is orthogonal. Six such mapped tiles can be patched together to cover the entire sphere. Because the edge of each tile is a coordinate line, shared with the neighboring tile, orthogonality ensures that the normal coordinate smoothly matches between tiles. The arrangement of these six tiles in computer memory is, as we shall see, relatively straightforward.

The parallelization of the MIT GCM uses a tile and halo strategy in which each tile might be associated with one processor in a multiprocessor computer. The halo region retains a duplicate of data on neighboring tiles, allowing the frequency of communication between processors to be reduced. The tiling strategy is readily reconfigured to implement a cubic topology.

Figure 1 shows one of many possible arrangements of six tiles that mimic the topology of the surface of a cube. This arrangement is similar to that of McGregor (1997) but is flipped in orientation. All pairs of tiles are treated identically. For convenience, we refer to the faces of the cube by colors: red (R), green (G), blue (B), magenta (M), yellow (Y), and cyan (C). The connectivity is such that interior edges in Fig. 1, namely R–G, G–B, B–M, M–Y, Y–C, and C–R, look as though the tiles are a regular decomposition of a plane. Most importantly, the coordinate systems smoothly match and are cooriented across these interior edges. However, the

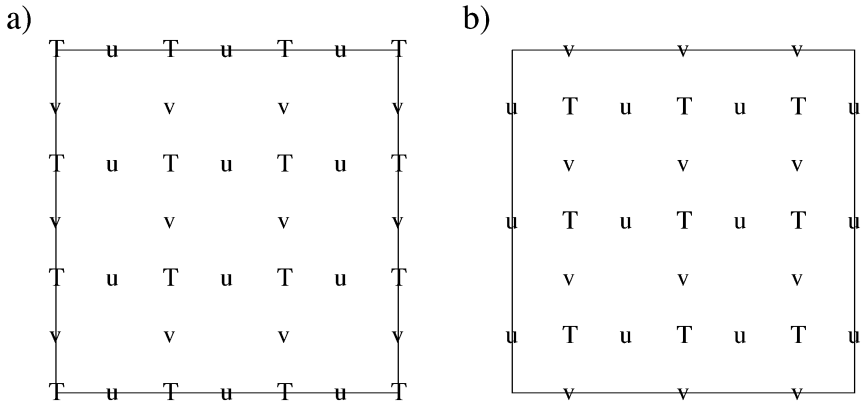

FIG. 2. The two possible alignments of a C grid with the tile boundaries (indicated by the square): (a) scalar points fall on the tile edge and corners, (b) normal flow components fall at tile edges.

connections across the upper outer edges, namely R–B, B–Y, and Y–R, require a rotation of coordinates. That is, the local coordinates and flow at the upper edge of the R tile must be rotated $\pi/2$ counterclockwise to match the local coordinates and flow of the B tile at the left edge, as is apparent by inspection of the 3D rendering of the tiles in Fig. 1. Such a transformation is not encountered in a regular decomposition of a plane. The vector transformation at the edges is implemented in the same low-level software used to update the halo regions. Thus, the numerical code is unaffected by the unusual topology.

The model variables are staggered in space using the Arakawa C-grid arrangement. There are two choices of alignment of the C grid with the tile boundaries, as shown in Fig. 2: (a) tracers and tangential flow components align on the tile edge, and (b) normal flow components align on tile edge. The second arrangement (b) is the most natural for ocean modeling and is the default in our model. We did not investigate choice a, but see no reason why it could not be implemented.

Regardless of whether arrangement a or b is used, there is a degeneracy at the corners of the cube associated with a singularity of the coordinate systems. One manifestation of this is illustrated in Fig. 3, where we have assumed arrangement b. Consider the interpolation of the v component of flow in the green coordinate system to the corner point (the interpolation is indicated by the solid line in the lower-right panel). The left-most contribution to the interpolation is in the halo region and ''virtually'' points to the red ''v.'' Because the green–blue coordinate systems match, the same interpolation should be replicated on the blue tile. Now, however, the (blue) halo v is set equal to $-u$ on the red tile due to the coordinate rotation on the red–blue boundary. Because the v and u on the red tile are separate, independent variables, the apparent value of v at the same geographic point is different on each of the blue and green tiles. This is clearly not sensible and must be rectified.

The only solution to this problem is to avoid interpolation of vector quantities to the cube corners. Discretizing the momentum equations on a C grid, whether in conservative (1) or vector-invariant form (2), inevitably places momentum fluxes or other vectors at both the tracer (cell centers) and vorticity (cell corners) points. This means that a different choice of grid alignment (e.g., Fig. 2a) does not solve the problem. However, as we now discuss, the vector-invariant form of the equations can be discretized using alignment Fig. 2b such that only scalar quantities fall on cell corners.

b. *Discretization of the vector-invariant equations on a grid with singularities*

The vector-invariant equations (7) can be discretized on a C grid using the most natural/simple forms except for the Coriolis terms. The components of the Bernoulli gradient $\nabla(K + \Phi)$ naturally aligns with u and v points, as does the vertical advection of momentum. The Coriolis terms involves interpolation of a component of a vector (the flow) and will be considered more carefully later. The model equations are discretized as follows:

$$\Delta p \Delta y_u [\Delta x_u \partial_t u - \overline{f + \zeta}^{j} \overline{\overline{\Delta x_v v}^{j}}^{i} + \delta_i(\Phi + K)] + \overline{\overline{\mathcal{A}\omega}^{i} \delta_k u}^{k} = \Delta p \Delta x_u \Delta y_u \mathcal{F}_u, \tag{8a}$$

$$\Delta p \Delta x_v [\Delta y_v \partial_t v + \overline{f + \zeta}^{i} \overline{\overline{\Delta y_u u}^{i}}^{j} + \delta_j(\Phi + K)] + \overline{\overline{\mathcal{A}\omega}^{j} \delta_k v}^{k} = \Delta p \Delta x_v \Delta y_v \mathcal{F}_v, \tag{8b}$$

$$\Delta \Pi \overline{\theta}^{k} + \delta_k \Phi = 0, \tag{8c}$$

$$\delta_i(\Delta p \Delta y_u u) + \delta_j(\Delta p \Delta x_v v) + \delta_k(\mathcal{A}\omega) = 0, \tag{8d}$$

$$\Delta p \mathcal{A} \partial_t \theta + \delta_i(\Delta p \Delta y_u u \overline{\theta}^{i}) + \delta_j(\Delta p \Delta x_v v \overline{\theta}^{j}) + \delta_k(\mathcal{A}\omega \overline{\theta}^{k}) = \Delta p \mathcal{A} Q, \tag{8e}$$

where

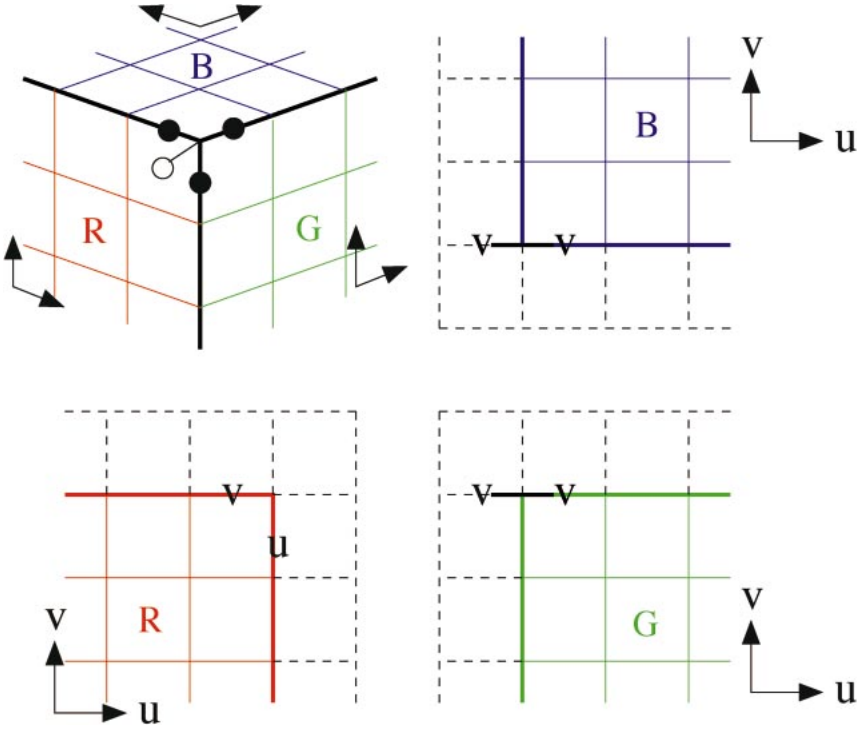

FIG. 3. Interpolating the v component of flow in the green coordinate system to the corner on the green tile should yield the same values as on the blue tiles because the coordinate systems are aligned. However, the green halo v is set equal to the red v, while the blue halo v is set equal to minus the red u. The solid stencil at top left indicates the locations of the flow components. The fourth point (empty circle) is the natural position for interpolating the v (in red–green coordinates) to the corner using values on the red tile.

$$\zeta = \frac{1}{\mathcal{A}_\zeta}[\delta_i(\Delta y_v v) - \delta_j(\Delta x_u u)] \quad \text{and} \tag{9}$$

$$K = \frac{1}{2}(\overline{u^2}^i + \overline{v^2}^j). \tag{10}$$

In the above equations, $\mathcal{A}$ (without a subscript) is centered on the tracer points. Otherwise, subscripts indicate centering on that subscript (i.e., $\mathcal{A}_\zeta$ is the area of a vorticity cell while A is the area of a tracer cell, and Δx_u is the distance between tracer points centered on u). We have chosen an enstrophy-conserving form for the Coriolis terms and an energy-conserving form for the vertical advection of momentum. These choices are not relevant to the discussion at hand but can have an impact on the smoothness of the solution (Sadourny 1975). The above discretization is second-order accurate on a regular grid, but accuracy may be reduced where a curvilinear grid has high curvature.

There are several issues of note in the above discretization.

1) FINITE-VOLUME DISCRETIZATION OF VORTICITY

Rather than discretizing the vorticity using finite differences

$$\zeta \approx \frac{1}{h_\phi \Delta y}\delta_i(h_\phi v) - \frac{1}{h_\lambda \Delta x}\delta_j(h_\lambda u), \tag{11}$$

we discretize using the finite-volume method. The vorticity is then expressed in terms of the circulation Γ,

$$\zeta \approx \frac{\Gamma}{\mathcal{A}}, \tag{12}$$

where

$$\Gamma = \oint v\, dl = \delta_i \int v\, dx - \delta_j \int u\, dy$$
$$= \delta_i(\Delta y v) - \delta_j(\Delta x u), \tag{13}$$

and $\mathcal{A}$ is the area enclosed in the circulation contour. In Fig. 4 it is clear that this area is six sided, with three circulation contributions. Note that in the finite-difference form (11) a higher-order representation widens the stencil of the difference operator in the direction of the differences. In contrast, a higher-order approximation of the line integral in Eq. (13) widens the stencil of interpolation (of the integrand) in the direction of the integral. Finite-difference and finite-volume methods, therefore, differ substantially at a higher order. At second order, the stencils of the finite-volume and finite-difference methods are the same in the interior of the

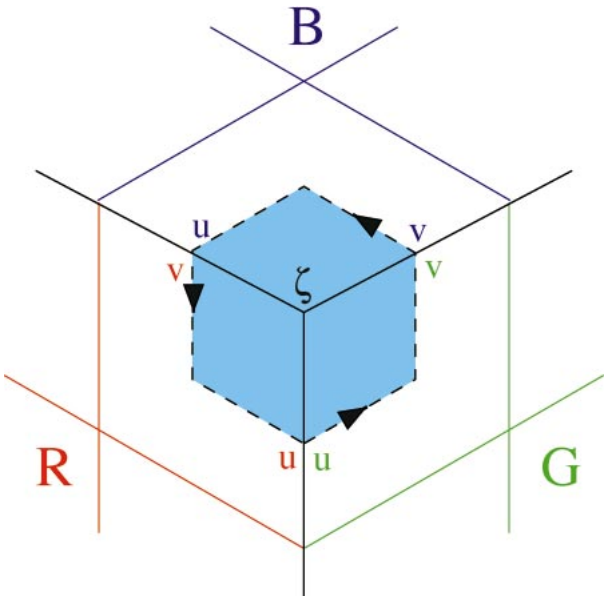

FIG. 4. The vorticity everywhere is discretized as the circulation divided by the area. At the tile corners the line integral involves six segments (three pairs of values) that uniquely defines the vorticity even though the coordinate singularity is enclosed within the area.

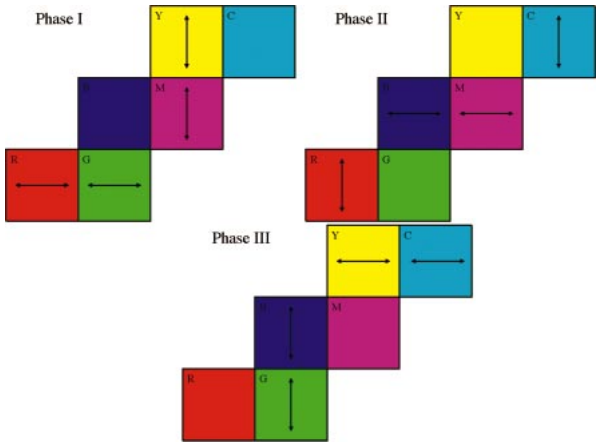

FIG. 5. Operator-splitting phases on the cube. Operations in a certain direction (meaning computational coordinate) are indicated by the arrows. Each *i*–*j* direction on each tile can be followed continuously around four sides of the cube so that operations in this direction should be coordinated. Three phases are required to cover both directions on all tiles. The sequence (I–II–III) is reversed (III–II–I) every other time step in the manner of Strang splitting.

tile, but differ numerically because $\mathcal{A} \neq h_\lambda h_\phi \Delta\lambda\Delta\phi$, as noted in section 2. Of most importance here is the fact that the finite-difference form yields ambiguous results at the corners, while the finite-volume form (using circulation) unambiguously defines the vorticity everywhere, even around coordinate singularities.

2) NONCOMMUTING DIFFERENCE OPERATORS

Finite-difference operators must be considered to not commute. That is, the order of operations *is* important; for example,

$$\overline{\overline{v}^{i}}^{j} \neq \overline{\overline{v}^{j}}^{i}. \tag{14}$$

Under normal circumstances the order of operations is not important, and such identities can be assumed. Now, however, we disallow such identities so that components of vector quantities can be interpolated ''around'' the singularities. For example, consider the linear (second order) interpolation of v to the u points, normally written as $\overline{v}^{ij}$. Writing the interpolation in two steps, we have two possible schemes:

$$v^* = \overline{v}^{i} \quad \text{and} \quad \overline{v}^{ij} = \overline{v^*}^{j}, \tag{15a}$$

or

$$v^\dagger = \overline{v}^{j} \quad \text{and} \quad \overline{v}^{ji} = \overline{v^\dagger}^{i}. \tag{15b}$$

In scheme (15a), v^* falls on the vorticity points and, thus, on tile corners and singularities. Hence, this scheme is subject to the degeneracy problem outlined above in section 3a. In scheme (15b), $v^\dagger$ falls at the cell centers. Here, there are no ambiguities and the operation is well defined. The final interpolation, evaluating $\overline{v}^{ji}$, is likewise unambiguous and well defined.

3) OPERATOR SPLITTING AND ADI-LIKE OPERATIONS

Operator splitting is used to accurately apply one-dimensional schemes in multiple dimensions (see Pietrzak 1998). For instance, we use flux-limited advection schemes to avoid false extrema (described in section 5a); these schemes work in one dimensions and are extended to three dimensions using operator splitting. In Cartesian coordinates operator splitting calculates fluxes in each direction consecutively, say *x* then *y*, and [in the case of Strang splitting (Strang 1968)] will reverse the order on each consecutive model step. If we apply such a scheme based on local coordinates within each tile, we find that fluxes on tile boundaries that are calculated independently on each tile do not match. This is because the local *x* and *y* directions are not globally contiguous across all tile boundaries. To overcome this, we implement operator-splitting methods using three phases, as illustrated in Fig. 5. During each phase, only four tiles are active and calculations are made in a direction pseudoparallel to a great circle that passes through those four tiles. The sequence of phases is reversed every other time step as in Strang splitting. This reduces biases in the operator-splitting error. However, some asymmetry remains in the system for the following reason: consider the fluxes evaluated during phase II. Fluxes in the M tile are evaluated in the local *x* direction using values previously updated during phase I, while fluxes in the B tile, evaluated in the local *x* direction, use unmodified values. Although the fluxes are contiguous across the B–M edge (i.e., conservation is satis-

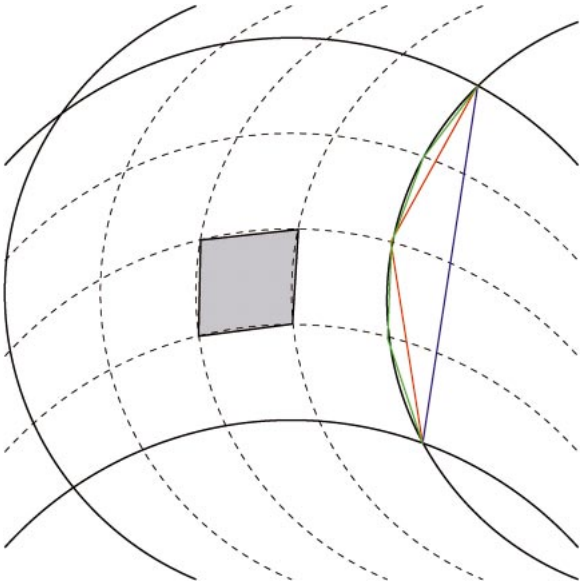

FIG. 6. To calculate the curvilinear lengths of the coarse grid (thick solid lines), a finer grid (dashed lines) is laid down and the lengths on the fine grid are approximated as great arcs. The blue line indicates the great arc approximation of the target grid. The two red segments are the great arcs used in a single refinement and the four green segments are those used in two refinements. The area of the grid cell is similarly approximated as the sum of the enclosed fine-grid cell areas, each approximated using the excess-area rule. This idealized figure illustrates how the refinement method rapidly converges on the true grid even in regions of high curvature.

fied), there is asymmetry in the evolution. Thus, operator splitting is not implemented perfectly. We overcome biases due to this error by permuting the order of phases every other model step (e.g. I–II–III, III–II–I, II–III–I, I–III–II, etc.).

4. Grid generation

In the discrete equations [(8)] we describe the grid in terms of length (Δx_u, Δx_v, Δy_u, Δy_v) and area ($\mathcal{A}$, $\mathcal{A}_\zeta$) only, and not in terms of map factors (h^u_λ, h^u_ϕ, ...). This allows the code to be written in a notation similar to that of a Cartesian grid model, which also happens to reduce the number of grid descriptors. Using metric factors, we would need two factors per point on the staggered grid, totaling eight factors. Using the conservative form of the momentum equations requires eight lengths and three areas, totaling 11 factors. In the above discretization of the vector-invariant equations only four lengths and two areas are needed, totaling six factors. These grid descriptors are only two-dimensional and, thus, are relatively cheap to store. Moreover, with fewer factors, consistency in grid generation is easier to maintain.

Calculation of the grid lengths and areas is carried out numerically. A fine grid is laid down such that the target grid can be constructed as a subset of the fine grid. Figure 6 shows an idealized, high-curvature grid cell with a fine grid overlaid, allowing for two refinements in a multigrid calculation. Curvilinear lengths on the surface of the sphere are approximated as great arcs (radius multiplied by subtended angle at center of sphere). Such an approximation is quite inaccurate if applied to the target (coarse) grid. However, approximating the curvilinear lengths as the sum of the fine-grid segments leads to an algorithm that rapidly converges to the true curvilinear length. The area of the grid cell is likewise approximated as the sum of the areas of the fine-grid cells. Each fine-grid cell is approximated as a quadrilateral of great arcs, the area of which is proportional to the ''excess'' (the excess of a quadrilateral x_s on the surface of a sphere is the sum of the internal angles α_i, minus 2π):

$$\mathcal{A} = r^2 x_s,$$

where

$$x_s = -2\pi + \sum_i \alpha_i.$$

The nodes of the fine grid are determined either by a semianalytical conformal mapping, as described in Rancic et al. (1996), or by numerically following conventional orthogonal grid–generation methods (via the solution of two elliptic equations for grid coordinates, see Fletcher 1991). Numerically generated grids tend to have some nonorthogonal traits; numerically evaluating the angles between intersecting grid lines reveals that the angle is never exactly $\pi/2$. We have found the original method of Rancic et al. (1996) to be superior in this regard with the least deviation from orthogonality, but nonorthogonality seems to be unavoidable in the immediate vicinity of the corner singularities.

The conformal mapping of Rancic et al. (1996) leads to a grid that has a near-uniform aspect ratio (of value unity) and with a ratio of longest-to-shortest element lengths that scales as $M^{1/3}$, where M is the number of elements on an edge. The shortest element length, therefore, scales as $M^{-4/3}$, which is significantly better than a conventional latitude–longitude grid (N^{-2}). However, this scaling behavior falls far short of the scaling of the gnomonic cubic grid (Sadourny 1972). For the gnomonic cubic grid, the ratio of longest-to-shortest element lengths rapidly approaches 2 as the number of points is increased. This means that the gnomonic grid scales perfectly in that the shortest length scales as M^{-1}. We, therefore, now consider how the conformal grid of Rancic et al. (1996) can be modified to improve the scaling.

We used two variants of a simple approach in which we rescale the coordinates after the original conformal mapping, thereby preserving orthogonality. For example, the tile coordinates (x_1, x_2) can be mapped to

$$(x_1', x_2') = \frac{1}{\tan^{-1}(2/3)}\left(\tan^{-1}\frac{2}{3}x_1, \tan^{-1}\frac{2}{3}x_2\right), \quad (16)$$

where the coordinates range from -1 to $+1$. This

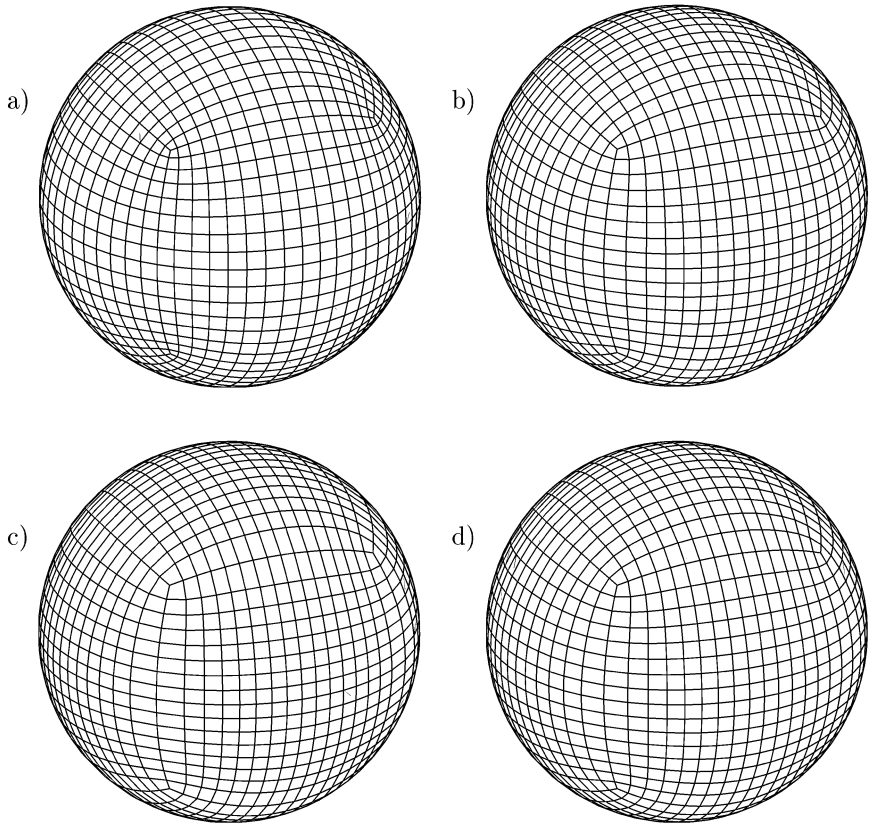

FIG. 7. Expanded cubic grids with 16 cells to a side (C16 for short) yielding 64 points around the equator. (a) The conformal projection proposed by Rancic et al. (1996). (b) As in (a) after a rescaling of coordinate following Eq. (16). (c) As in (a), but numerically rescaled such that the edge segment lengths are uniform. (d) As in (c), but applied to the third grid line.

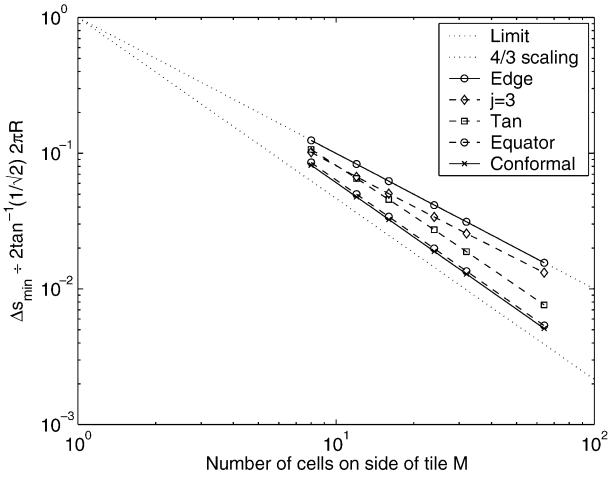

FIG. 8. The scaling behavior of a variety of cubic grids. Plotted is the shortest grid element Δs_{min}, normalized by the length of a cube edge. (Magenta) is the conformally mapped grid of Rancic et al. (1996), labeled ''Conformal,'' (red) using the remapping in Eq. (16) labeled ''tan,'' (blue) has equal element lengths along an edge, labeled ''Edge,'' (cyan) has equal element lengths along the equator, labeled ''Equator.'' For comparison, $N^{-4/3}$ and N^{-1} are plotted as dotted curves. The N^{-1} scaling represents the best scaling behavior possible.

stretches the coordinate near the edges and compresses them near the tile center. The resulting grid is shown in Fig. 7b, and the scaling behavior is shown in Fig. 8 (red curve labeled ''tan''). It has a more uniform distribution of segment lengths along a tile edge, but at the expense of symmetry (cf. Figs. 7a and 7b); the aspect ratio of cells near the tile edges is no longer as close to unity as for the Rancic et al. (1996) grid.

Another rescaling is made numerically by finding the function that makes the segment lengths along the edge exactly equal (and, therefore, correspond to the gnomonic grid but only on the edges). The grid is shown in Fig. 7c. Although the distortion of the aspect ratio is now severe, this grid has the same perfect scaling as the gnomonic grid; the smallest length scales as M^{-1}. Geometrically this can be seen because the smallest elements appear at the tile centers, but cannot be made shorter than the edge elements without a convergence of coordinate lines. The other extreme has elements of exactly equal lengths along a great arc across the center of a tile (cyan curve labeled ''equator''). This compresses the resolution at the corners similar to the original conformal mapping. A compromise is to chose a coordinate line near the edge of the tile and set lengths

to be uniform along that line. The result of choosing a line $x_1 = 3/8$, the third grid line from the edge, is shown in Fig. 7d, and scales as shown by the green curve labeled "$j = 3$".

5. Numerical results

To test the implementation of the model on the cubed sphere we first conducted several of the shallow-water tests described by Williamson et al. (1992). These were especially useful in revealing implementation problems. The three-dimensional model is tested by repeating the experiments of Held and Saurez (1994, hereafter HS94). Not all results are reported here—we have selected those that we found most useful in identifying strengths and weaknesses of the model.

a. Williamson test 1: Advection of cosine bell

A passive tracer is advected around the sphere by a specified flow field equivalent to solid body rotation around an arbitrary axis. An initial tracer distribution (a cosine bell on the equator of the flow rotation) is advected around the sphere once in exactly 12 days. The flow is nondivergent and so can be described by a streamfunction $\psi(\lambda, \phi)$ given by

$$\psi(\lambda, \phi) = -aU_o(\sin\phi \cos\alpha - \cos\lambda \cos\phi \cos\alpha),$$

where α is the angle between the geographic coordinate axis and the axis of the solid body rotation; $U_o = 2\pi a/(12 \times 86\,400) \sim 40$ m s^{-1} is the maximum wind speed, and $a = 6370$ km is the radius of the sphere.

The coarsest-resolution grid used was C32 (the cubic grid of Rancic and Purser with 32 points across a tile), which is equivalent to G64 (128×64 points in spherical polar coordinates) in equatorial resolution. The calculations are made on grids of different resolutions to establish the rate of convergence. Here we used C32, C46, C64, and C96, all using the conformal cubic grid of Rancic et al. (1996), shown in Fig. 7a, which reflect resolutions at which this model is often used.

The model can be configured to use one of many advection schemes. Here, we show three variants of third-order interpolation (based on a uniform mesh). Figure 9a shows the solution after 12 days on a C96 grid using third-order spatial interpolation with a second-order Adams–Bashforth scheme; dashed contours are the true solution. This formally reduces to second order for variable resolution grids. The error is shown in Fig. 9b and the time evolution of the l_1, l_2, and l_∞ norms as defined by Williamson et al. (1992) are plotted in Fig. 9c. The better-than-second-order convergence is shown in Fig. 10a (circles). The second variant uses a direct space–time discretization using a third-order polynomial representation (Hundsdorfer and Trompert 1994; Hundsdorfer et al. 1995; Pietrzak 1998), which again formally reduces to second order on a variable space mesh. The error (not shown) and its scaling (crosses in Fig. 10a) are much the same as for the linear advection scheme. Figure 9d shows the error using a Sweby flux limiter on the direct discretization fluxes (Hundsdorfer et al. 1995). Note that the errors are spatially more compact (due to the limiting), but have more structure due to the multidimensional aspect of the algorithm.

b. Impact of different cubic grids

We motivated the generation of new grids by rescaling the tile coordinates (see section 4) as a means of improving the scaling of the grid. We now evaluate these grids by finding the maximum time step that yields a "stable" solution in the Williamson et al. (1992) experiment 1, as described in section 5a.

The grids shown in Figs. 7a, 7b , and 7c should allow successively larger time steps. For each of these grids we increased the time step in increments of 1 min until the solution either became unstable or began to exhibit visible noise at the grid scale. Using the upwind-biased third-order interpolation with the second-order Adams–Bashforth scheme, we find that we can indeed achieve longer time steps on the rescaled grids ("tan" and "edge," Table 1). However, the maximum Courant number ($\mathrm{CFL}_{\max}$) for which the solutions are stable are reduced on the rescaled grids. We believe that this is due to the nonuniform aspect ratio.

This hypothesis is borne out further if we use a more stable advection scheme; the third-order direct space–time method (Hundsdorfer and Trompert 1994; Hundsdorfer et al. 1995; Pietrzak 1998) is stable up to Courant number 2 and increases accuracy as the Courant number approaches 1. Table 2 shows the maximum stable Δt and associated Courant numbers. We now find little difference in Δt between the grids, but with a wider range of stable Courant numbers.

It seems, then, that rescaling the tile coordinates yields grids with more uniform spacing, thus, allowing longer time steps in models. However, the improvement is limited by the effects of anisotropic resolution or distorted aspect ratios.

We evaluated these grids in the Williamson et al. test 1 case of section 5a. Figure 10 shows the convergence rates for differently rescaled grids all using the same Courant number and linear advection scheme. The equator grid (with equal element lengths around the equator) is visually very similar to the original conformally mapped grid of Rancic et al. (1996), and the numerical convergence rates are indistinguishable. Likewise the edge grid and tan grid are quite similar and also scale in a similar fashion to each other. The tan rescaling does have systematically reduced l_2 norms (vertical shift of dashed line), but the scaling is only slightly better (slope of dashed line is 2.7 while slope of solid line is 2.6).

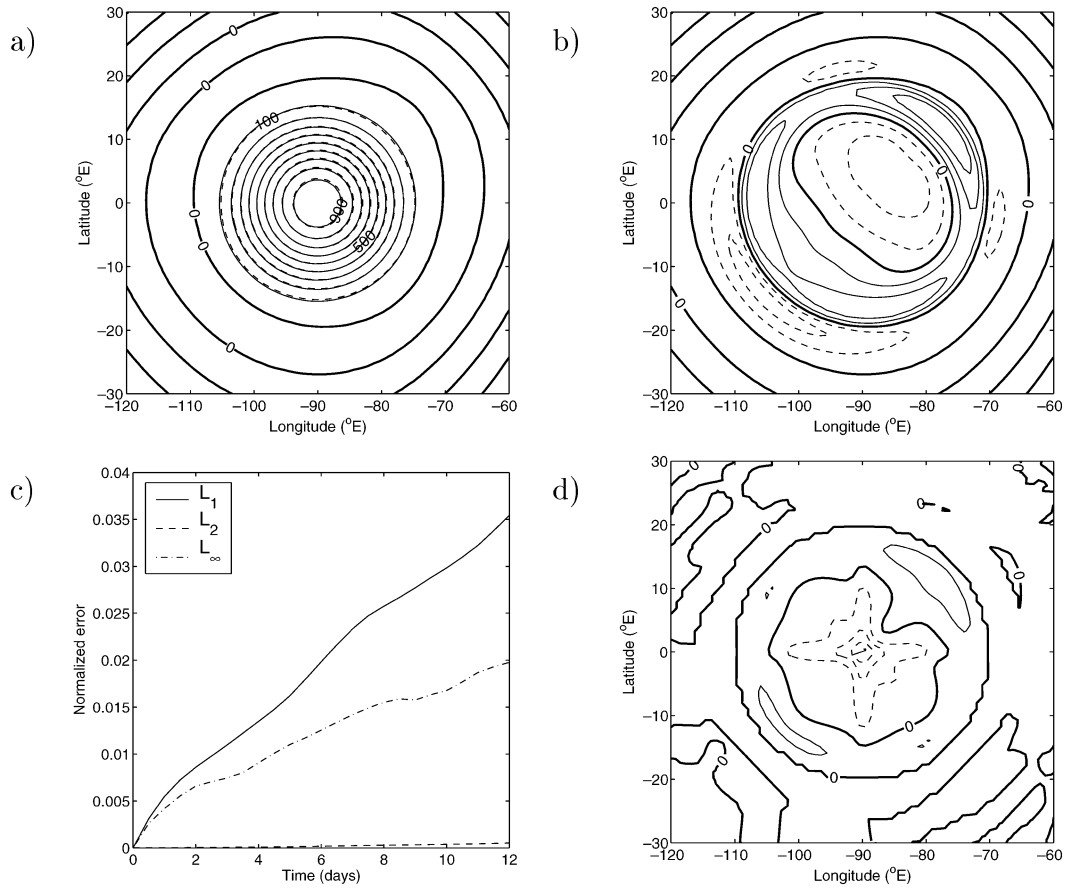

FIG. 9. Results for Williamson et al. (1992) test 1 at a flow angle of $\pi/4$ that translates the tracer anomaly directly over the grid singularities. (a) Solution after 12 days obtained on a C96 conformal grid using linear differencing and (b) the error (with contour interval of 5 m). (c) The evolution of the l_1, l_2, and l_∞ norms as defined by Williamson et al. (1992). (d) The error using nonlinear interpolation (direct space–time discretization) with the Sweby flux limiter.

c. Held and Suarez forcing

The experiments of HS94 were reproduced using the same model on both a conformal cubic grid (C32) and the conventional geographic latitude–longitude grid (G64), which are of equivalent resolutions but are slightly coarser than those used by HS94. The forcing is as described by HS94 but cast in pressure coordinates.

We use 20 equal thickness levels in the vertical, which has enough resolution to represent the gross structures seen in the HS94 solutions. Despite the moderately low horizontal resolution, the climatology produced by the model is quite reasonable, as will be described next.

The G64 model requires zonal filters applied poleward of $\pm 45°$ for numerical stability. Both models use an eighth-order Shapiro filter on both temperature and momentum tendencies to represent small-scale dissipation. The time step using G64 is 7.5 min and that using C32 is 10 min. Note that internal gravity waves are explicit so that even though advective CFL limits would allow a time step of the order of 1 h, the internal gravity wave speed limits the time step in the full dynamical model. Note, also, that we are using the conformal grid of Rancic et al. (1996); the maximum-allowed time step on the C32 using the tan rescaling gives a time step of 12.5 min.

The zonal mean climatologies (10-yr means) are shown in Fig. 11. There is broad agreement between both the G64 and C32 solutions and the results of HS94. However, on closer examination the G64 solutions have some notable discrepancies, namely a high-altitude polar jet (Fig. 11e), probably associated with the strong temperature gradients near the poles (Fig. 11c) and a lack of tilt in the midlatitude jets. The C32 solutions show no such problems and compares more closely with HS94. We presume that the primary reason for the differences between G64 and C32 are due to the zonal filters. It is not clear why our implementation of zonal filters produces significantly worse results than G72 of HS94. HS94 point out that G72 was the coarsest resolution that produced agreeable results, so the problem may lie with the resolution.

Figure 12 shows the zonally averaged eddy variance of temperature $\overline{T^{*}}^{xt}$. Again, C32 is more favorable than G64; there is more eddy activity in C32 again, we believe, due to the absence of zonal filters in C32.

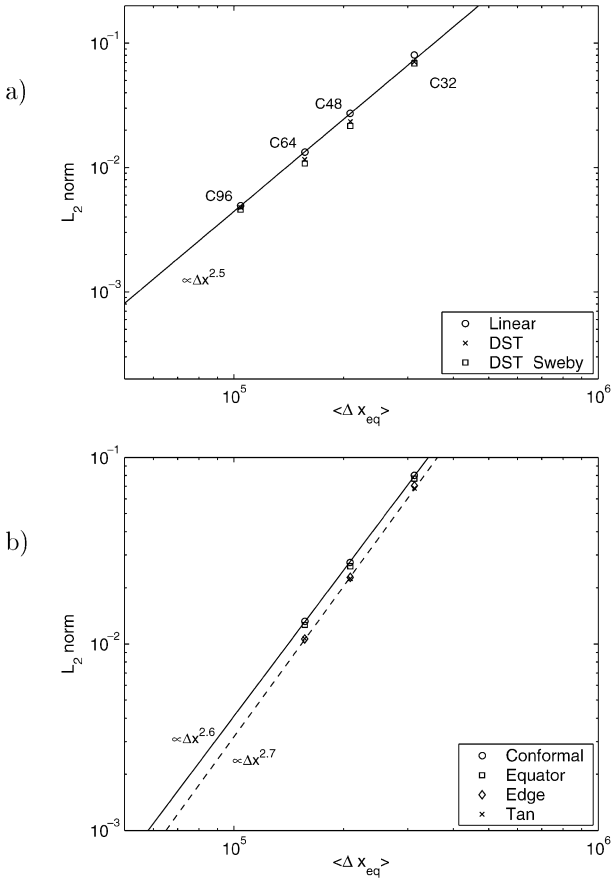

FIG. 10. (a) The scaling of the l_2 norm with resolution on the conformally mapped grids using the same Courant (CFL) number: linear interpolation (circles) nonlinear (DST; crosses), and DST with the Sweby flux limiter (squares). The scaling is approximately $O(\Delta x^{2.5})$ for the linear scheme; on a regular grid this scaling would be $O(\Delta x^3)$. (b) The scaling of l_2 norm for different rescaled grids: conformally mapped grid of Rancic et al. (1996; circles) also shown in (a), rescaling to give uniform resolution along the equator (squares); rescaling to give uniform element lengths along the edge (diamonds); the analytic rescaling given by (16) (crosses).

Zonal averages are useful for comparing climatologies but might hide problems associated with the underlying horizontal grid. For example, standing patterns might be triggered by the variable resolution around a latitude circle. To reveal such problems we also examine the horizontal distributions of quantities such as eddy temperature variance. Figures 12c and 12d show the time average eddy variance at 950 mb. We have overlaid edges of the cubic grid for reference. The pattern is clearly dominated by a latitudinal profile, but does reveal zonal structures. Subtracting the zonal mean (Figs. 12e and 12f) reveals these variations to be of order 2 K^2 in both models, with no obvious correlation to the cubegrid in C32. The amplitude of these anomalies should decay as the length of time-averaging period is increased; that is, the statistics should be invariant with longitude if enough samples

TABLE 1. The maximum Δt and corresponding CFL number that could be used and still produce a stable and smooth solution. The grids were all of size C32 and the advection scheme was the third-order upwind interpolation used in section 5a.

Scheme	Δt_{max}	CFL_{max}
Conformal (Fig. 7a)	38 min	0.68
Tan [Eq. (16), Fig. 7b]	50 min	0.6
Edge (equal edge elements, Fig. 7c)	51 min	0.47

are made. The amplitude of $\overline{T'^2}^t - \overline{T'^2}^{xt}$ is plotted in Fig. 13. Unfortunately, the decay is slow. Thus, to discern any deviation from expected behavior require a significantly longer integration than our resources allow. This serves to indicate both that our statistics are indifferent to the underlying grid, but also that they are perhaps insufficient to detect biases due to the underlying grid.

Another diagnostic given by HS94 was the eddy zonal spectra. We show the spectra for both T^{*2} and U^{*2} (Fig. 14). The former has clear peaks at mode 6 in both G64 and C32. The eddy energy spectra show similar structures to those from HS94. If, at this coarser resolution, we had found a peak at mode 4, we most certainly might have suspected some influence of the underlying cubic grid.

d. Global ocean circulation

As mentioned, the MIT GCM can model both the atmosphere and ocean. Here we demonstrate that the ocean circulation found using the cubic grid is comparable to the solution found using the spherical polar grid.

The model is configured with 15 levels in the vertical with a maximum depth of 5200 m, forced with monthly wind stress from Trenberth et al. (1989), observed heat and freshwater fluxes from Jiang et al. (1999), and with a restoring of sea surface temperature to climatology by Levitus and Boyer (1994) with a time scale of 60 days. The model parameters are listed in Table 3. Calculations are made on a spherical polar grid with 4° resolution (G45) and on a comparable cubic grid C24, which has nominally 3.75° resolution. The G45 calculation uses a typical form of viscous stress tensor to represent the eddy viscosity. However, on the expanded spherical cube, a stress tensor places vector quantities on the singularities. To avoid this we use the vorticity-divergence form of viscosity that does not satisfy varies higher-

TABLE 2. As in Table 1, but using the direct space–time third-order discretization with operator splitting.

Scheme	Δt_{max}	CFL_{max}
Conformal (Fig. 7a)	122 min	2.1
Tan [Eq. (16), Fig. 7b]	119 min	1.4
Edge (equal edge elements, Fig. 7c)	120 min	1.13

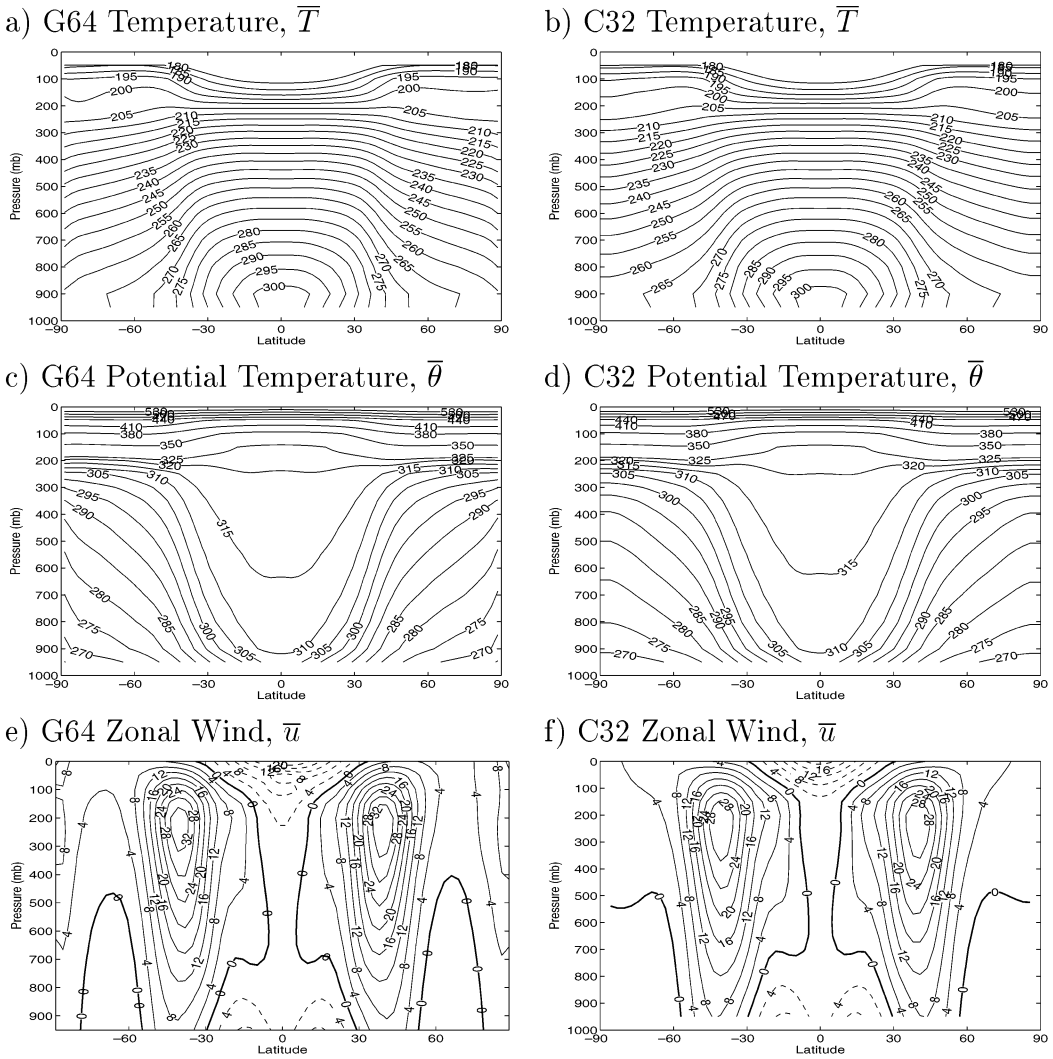

FIG. 11. Solutions from G64 and C32 calculations, both using five vertical levels driven by HS94 forcing. All fields are time and zonal means. (top) In situ temperature $\overline{T}$ for (a) the conformal expanded cube grid C32, (b) the regular spherical polar grid G64, (c), (d) potential temperature $\overline{\theta}$, and (e), (f) zonal winds. Results are comparable to those of HS94. The G64 solutions have significant problems in the highest latitudes where zonal filters are in use; strong temperature gradients and a polar jet appear right at the poles. The C32 model appears to treat the high latitudes much better and is closer to the G72 calculation of HS94.

order properties that an eddy viscosity should (Griffies 2004).

The bathymetry was generated from the earth topograph 5-min (ETOPO5) world bathymetry using a topology-preserving algorithm described by A. Adcroft (2004, unpublished manuscript). We did not hand edit the data (e.g., deepen overflows or widen passages), but did fill in the Arctic, north of 80°N in G45 to avoid the converging meridian problem. Despite using an objective interpolation method, the resulting topographies can be quite different simply because the different horizontal grids place features at different locations. The differences in topography between the two models is apparent in the land mask shown in Fig. 15.

Figure 15 shows the drift in potential temperature from the Levitus (2004) initial conditions at 290-m depth after 500 yr of integration as a 100-yr time mean. Broadly speaking, the drifts appear to be similar in the Tropics and Northern Hemisphere. The largest differences appear in the Southern Hemisphere. This is most likely due to the sensitivity of the Antarctic Circumpolar Current (ACC) dependence on the geometry of the Drake Passage and underlying topography. Most importantly, there are no apparent correlations with the nodes of the cubic grid.

The global Eulerian mean meridional overturning circulations (MOC) for G45 and C24 are shown in Fig. 16. The calculation of MOC in C24 involves integrating the

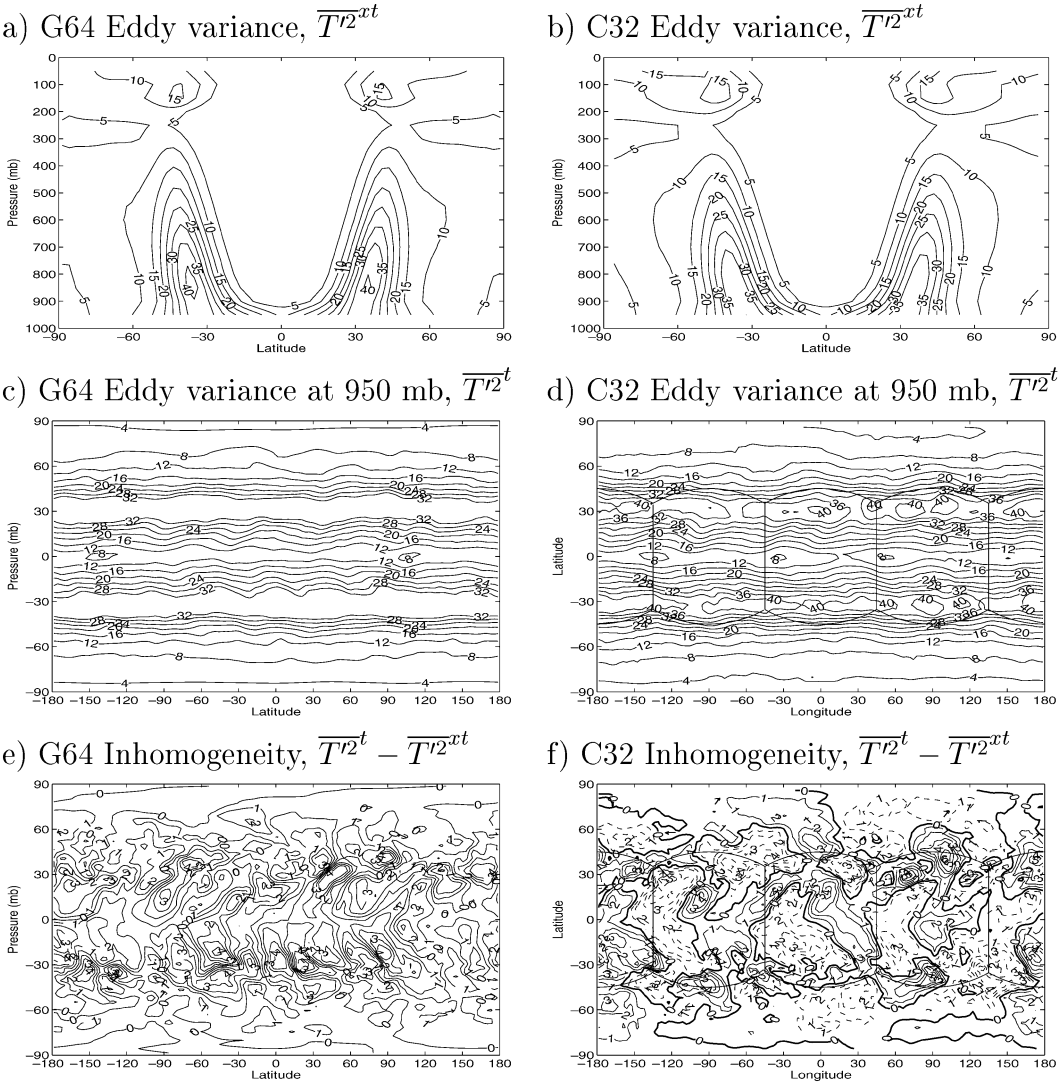

FIG. 12. The zonal time mean temperature variance $\overline{T'^2}$ in the (a) G64 and (b) C32 models. There is clearly more eddy activity in the C32 model although not quite as much as the HS94 G72 model. This is most likely due to the lower horizontal and vertical resolutions. The reduced eddy activity in the G64 model is probably due to the zonal filters that begin to damping motions at $\pm 45°$ latitude. (c), (d) The time mean variance $\overline{T'^2}^{t}$ at 950 mb over 10 yr. Because the forcing is zonally symmetric, the solution and eddy statistics should approach zonality. (e), (f) The same variance as (c) and (d) with the zonal mean removed. If inhomogeneities or discontinuities in the grid are influencing the solution we expect to see correlations with the underlying grid.

normal flow across zigzag lines that match the model grid but stay close to latitude circles. The overturning streamfunctions shown are broadly similar with a Northern Hemisphere maximum of 28.7 Sv (1 Sv $\equiv 10^6$ m^3 s^{-1}) that differs by only 0.1 Sv between the models. The strength of the deep overturning of Antarctic Bottom Water (AABW) is different; where the G45 model has 9 Sv, the C24 model has only 4.5 Sv. This may be due to the different resolution in the Weddell Sea (80°S, 50°E), as is apparent in Fig. 15.

Figure 17 shows the zonal time average salinity in G45 and C24. Again, the models broadly agree but differ in some details; the denser patch of North Atlantic Deep Water in G45 and fresher surface waters around Antarctica. Such differences in buoyancy easily lead to changes in deep circulation observed above and themselves may depend on subtle details of the model configuration. Again, we suspect that the different topography is largely the cause.

The inhomogeneities introduced by topography makes assessing the influence of the horizontal grids in ocean models very difficult. However, based on the broad agreement of the solutions, we conclude that the use of the cubic grid in ocean models in quite appro-

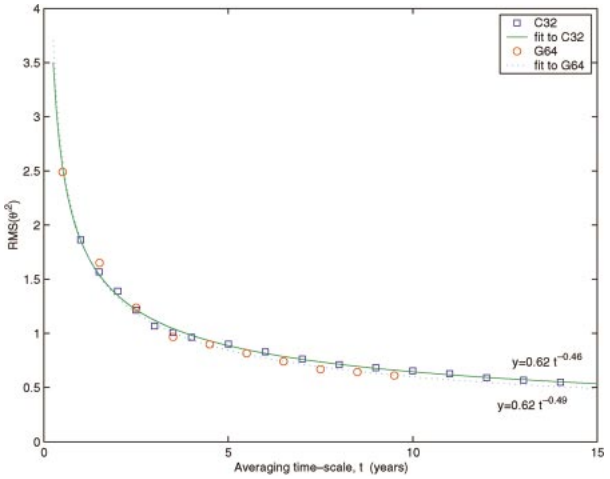

FIG. 13. The amplitude of the inhomogeneity $\overline{T^{*2}}^{t} - \overline{T^{*2}}^{xt}$ as a function of averaging time window. This reflects the amplitude of patterns such as in Figs. 12e and 12f; $\overline{T'^2}^{t}$ should become zonally uniform as the time window for averaging is made longer. C32 (squares) and G64 (circles). Even after 15 yr of averaging, the inhomogeneities are dominated by random fluctuations inherent in the dynamics and not by features correlated to the grid.

priate, and even beneficial, because the Arctic ocean can be treated properly.

6. Discussion

We have described the implementation of a finite-volume hydrodynamical kernal on a general orthogonal curvilinear grid. The use of vector-invariant rather than conservative momentum equations avoids the need to write out the general metric terms for an arbitrary grid. By restricting ourselves to orthogonal coordinates we were able to adapt an existing model; a general curvilinear, nonorthogonal coordinate model would require substantially more steps and algorithmic changes.

The conformal cubic projection of Rancic et al. (1996) was used for gridding the sphere. This avoids the converging meridians of conventional geographic coordinate grids. The cubic grid has eight coordinate singularities at the corners of the projected cube. The singularities pose a difficulty for discretization. By using the finite-volume interpretation of dependent variables we are able to replace gradients across the singularities with line integrals around an enclosing area. We also avoid interpolation across a singularity by choosing a particular order of operations so that interpolated terms are always expressed in terms of intermediate variables that never fall at the corners.

On the grid of Rancic and Purser, the shortest grid length scales as $M^{-4/3}$ (M is the number of elements along a tile edge), while a spherical polar grid scales as N^{-2} (N is number of points between poles). The $M^{-4/3}$ scaling is inferior to the perfect M^{-1} scaling of the gnomonic cubic grid (which is nonorthogonal). We were able to generate other orthogonal grids simply by rescaling the coordinates. One of these grids also has a M^{-1} scaling, but the improved distribution of resolution always comes at the expense of uniform aspect ratio. We demonstrated that the CFL criteria improve for the rescaled grids. Use of these grids will most likely depend on the particular application.

One concern in implementing the model on the cubic grid was that it is orthogonal everywhere except at the cube corners; only three cells meet at the corner node and so the angle between coordinate lines on the tangent plane is $(2/3)\pi$ and, thus, is not orthogonal. This happens only at a singularity so one might hope that the dynamics only feels a ''pin prick'' associated with nonorthogonality. To deal with nonorthogonality at the corners, the model equations could be written in general curvilinear coordinates, essentially adding cross terms to the orthogonal coordinate equations. On the other hand, our approach is not formally ''missing any terms'' because we do not discretize terms at the singular points. Higher curvature and finer resolution is also associated with the cube corners compared to elsewhere on the grid. The truncation errors in the model must, therefore, possess a spatially dependent envelope. Such inhomogeneities could affect the local dynamics and might show up in long-term statistics. However, as shown here, the solutions seem unaffected by the variable resolution and the special nature of singular points. This suggests that our finite-volume treatment at singularities is successful.

Solutions obtained with the HS94 forcing are very encouraging. One possible explanation for not seeing stronger effects of the grid is that the model is not accurate enough. Perhaps at a higher resolution variable resolution and singularities might manifest themselves more strongly. We expect that this will not be the case because we saw no such indication in the Williamson et al. (1992) tests. However, we have not yet made higher-resolution runs with the HS94 forcing.

We have not presented atmospheric calculations here in which zonal symmetry is broken by the forcing (or orography). The primary reason is that we have focused on seeking asymmetries or structures associated with the grid. We are confident that the model is accurate enough that such grid artifacts are not evident using 15 yr of statistics. Calculations with orography and realistic forcing are now being made. The impact of the grid will be assessed by changing the orientation of the grid geographically by, for example, zonal shifts. Unfortunately, by shifting the geography we are forced to change the orography and forcing because the interpolation and variable resolution of the grid leads to different numeric representations. It is, therefore, significantly more complicated to assess such grids in a realistic configuration. This will be the subject of future work.

Assessing the cubic grid in the ocean confronted this same problem of evaluating a nonsymmetric solution. The broad similarity between solutions on the cubic and geographic grids suggests that the approach is viable.

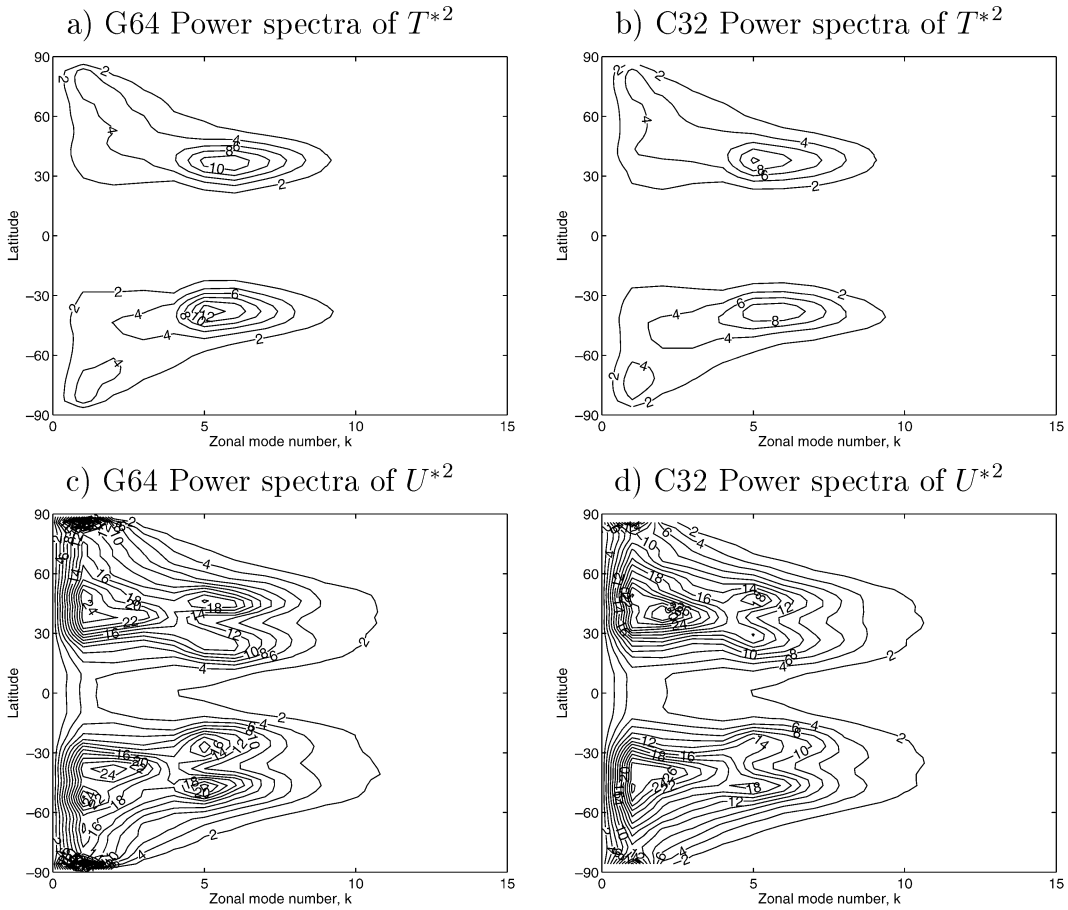

FIG. 14. Spectral power of eddy temperature variance T^{*2} as a function of zonal wavenumber in the (a) G64 and (b) C32 models following HS94. Both models have peaks at mode 5, slightly enhanced in C32. Note the absence of power at highest latitude in G64 due to the zonal filters whereas there is variability at all points on the C32 grid. Power spectra of eddy zonal winds U^{*2} for (c) G64 and (d) C32 models.

However, one potential criticism levied against the use of cubic grids is that the resolution is too uniform; the smaller deformation radius in the ocean at high latitudes warrants finer resolution near the poles. The cubic grids presented here used the same number of elements on each tile, yielding near-uniform resolution. Projecting a tall box onto the sphere can produce equivalently uniform resolution around the equatorial tiles and refined resolution at the poles. Thus, there is unexplored flexibility in the approach outlined here that could be used to great effect to model the ocean.

An awkward issue that arises when implementing an ocean model on the expanded spherical cube is the treatment of eddy viscosity. Formally, the eddy stress tensor should satisfy certain higher-order constraints such as to conserve total angular momentum (Griffies 2004). However, the use of a stress tensor with nonzero off-diagonal elements places vector quantities on the grid singularities, which, as we have explained, is degenerate. We avoided this here by not using a stress tensor but this is not completely satisfactory; the problem of satisfying higher-order constraints on such grids as this needs to be examined more closely.

We discussed the advantageous scaling of the expanded spherical cube with respect to the conventional geographic grids. The $N^{4/3}$ scaling is a huge improvement over the N^2 scaling. In global ocean modeling, an alternative solution to the converging meridians problem is to twist the Northern Hemisphere grid so that the polar singularity falls within land. Another alternative is the tripolar grid,[1] where the Mercator grid is cut off at some latitude (say 80°N) and another grid is embedded within that latitude circle. Again, the two singularities that fall on the cut-off latitude circle can be oriented so as to fall in land. Both approaches hide the singularities and the associated fine resolution in land and thereby avoid much of the problem. These grids scale better than the conformally expanded spherical cube and remain competitive at a high resolution.

[1] The grid has three singularities or poles, but is sometimes referred to as the bipolar grid.

TABLE 3. Parameters used in the ocean circulation experiments.

Horizontal eddy viscosity A_h	3×10^5 m^2 s^{-1}
Vertical eddy viscosity A_v	1.67×10^{-3} m^2 s^{-1}
Bottom drag coefficient C_D	1×10^{-3} s^{-1}
Isopycnal/thickness eddy diffusion κ_{GM}	10^3 m^2 s^{-1}
Vertical eddy diffusion κ_v	5×10^{-5} m^2 s^{-1}
Enhanced mixing (convection) κ_c	10 m^2 s^{-1}
Reference density ρ_o	1035 kg m^{-3}
Level thickness Δz (m)	50, 70, 100, 140, 190, 240, 290, 340, 390, 440, 490, 540, 590, 640, 690

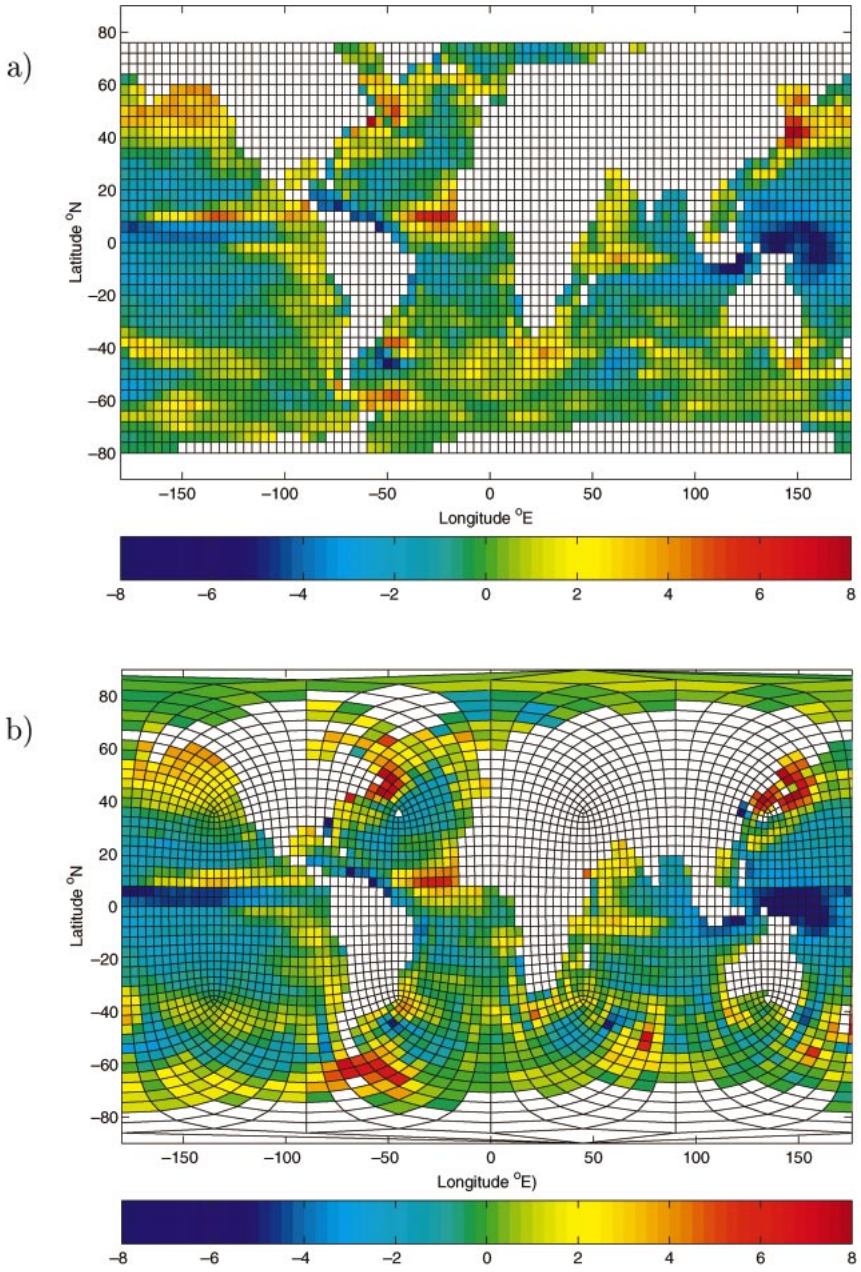

FIG. 15. The 100-yr mean potential temperature drift from Levitus (2004; °C) at 290-m depth after 500 yr of integration. The largest differences between the two models (a) G45 and (b) C24 appear in the Southern Ocean.

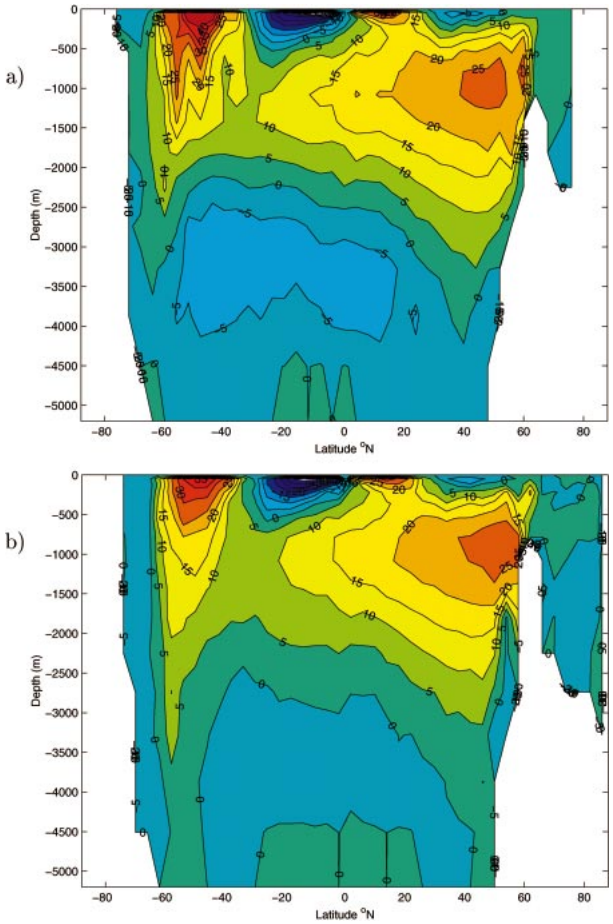

FIG. 16. The global Eulerian mean meridional overturning streamfunction ψ (Sv) after 500 yr of integration. The overturning in (a) G45 is typical of OGCMs forced with freshwater and no restoring of salt. The overturning in (b) C24 is calculated by integrating the net normal flow across zigzag lines that matches the grid. The overturning is broadly similar to that in G45, but note the stronger deep circulation of Antarctic bottom water in G45.

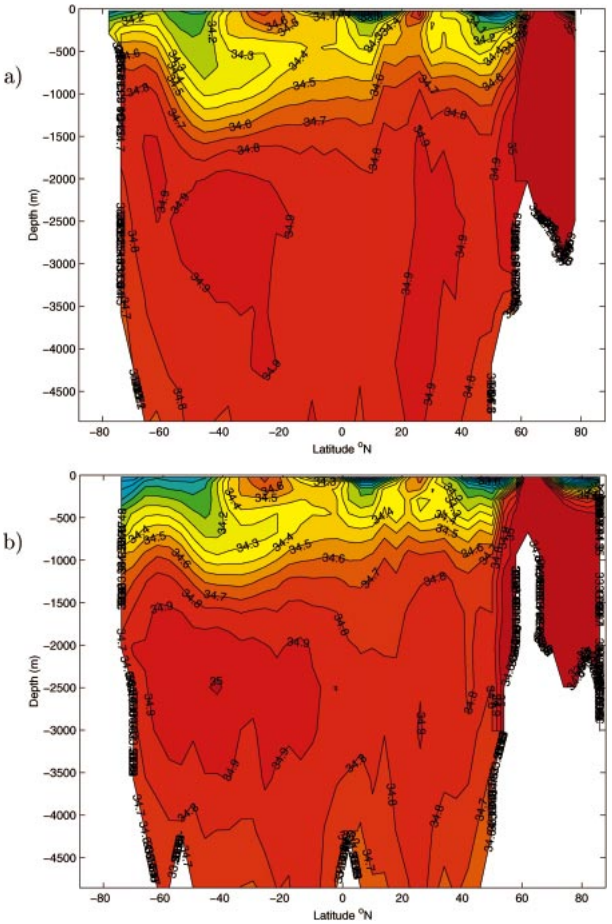

FIG. 17. The zonal time average salinity (psu) after 500 yr of integration. The average salinity in (a) G45 and (b) C24 are broadly similar but the North Atlantic Deep Water in G45 is slightly denser.

However, they are not options for atmospheric modeling. In the context of coupled models, it is more natural to couple models on the same or related grids; using the cubic grids for both atmosphere and ocean greatly simplifies the interpolation stages in the coupling process.

Although the scaling of the orthogonal cubic grids is an improvement over the conventional geographic grids, they do not scale as well as the gnomonic cubic grid or grids based on the icosahedra (Ringler and Randall 2002). These cannot be used in an orthogonal coordinate quadrilateral element model. However, both types of grids could be adopted if the limitation to orthogonal coordinates were dropped. For resolutions employed in ocean modeling today (around ⅛°) the conformally mapped expanded cube is competitive, but to achieve much higher resolutions in the future may require a general coordinate approach or another novel method to be used.

APPENDIX

a. General curvilinear coordinates

Sadourny (1972) and Rancic et al. (1996) examined solutions obtained with the shallow-water equations on the gnomonic cubic grid. The coordinate system here is not orthogonal and the model equations must be written in terms of general coordinates. Written in tensorial form, the shallow-water equations are then

$$\partial_t u_i + \epsilon_{ij} G(f + \zeta) u^j + \partial_{x^i}\left(\phi + \frac{1}{2} u_k u^k\right) = 0, \quad \text{(A1a)}$$

$$G\partial_t \phi + \partial_{x^i}(G\phi u^i) = 0, \quad \text{(A1b)}$$

where u_i and u^i are the covariant and contravariant components of the flow, ϕ is geopotential, $G = |g|^{1/2}$ is the square root of the determinant of the metric tensor

g_{ij}, and ϵ_{ij} is the permutation operator. The metric tensor is given by

$$g_{ij} = \mathbf{e}_i \cdot \mathbf{e}_j \qquad i, j = 1, 2,$$

where $\mathbf{e}_1$ and $\mathbf{e}_2$ are the base vectors of the coordinate transformation. The vorticity ζ is given by

$$\zeta = \frac{1}{G}\epsilon^{ij}\partial_{x^i}u_j.$$

The covariant and contravariant components are related by the metric tensor

$$u_i = g_{ij}u^j, \qquad u^i = g^{ij}u_j,$$

where

$$\begin{pmatrix} g^{11} & g^{21} \\ g^{12} & g^{22} \end{pmatrix} = g^{ij} = g_{ij}^{-1} = \frac{1}{G^2}\begin{pmatrix} g_{22} & -g_{12} \\ -g_{21} & g_{11} \end{pmatrix}.$$

A more complete description of general curvilinear coordinates can be found in Fletcher (1991).

There are two key algorithmic differences between general curvilinear and orthogonal coordinate models: (i) nonorthogonality introduces cross terms in the equations that can significantly affect the algorithm and performance, and (ii) the equations in general curvilinear coordinates are expressed in terms of covariant and contravariant flow components that introduce new steps in the algorithm compared to the algorithm in orthogonal coordinates.

Associated with the appearance of new terms in general curvilinear coordinates, relative to the orthogonal coordinate model, is the need for new steps in the algorithm. For example, the contravariant components need to be evaluated. This is straightforward when the flow components are collocated, such as on an Arakawa A or B grid, but on a C grid we have to interpolate the components in space as follows:

$$u^1 = g^{11}u_1 + g^{12}\overline{u_2}^{ij}, \tag{A2a}$$

$$u^2 = g^{21}\overline{u_1}^{ij} + g^{22}u_2. \tag{A2b}$$

We can expect that the spatial averaging permits a computational mode to exist and be problematic, especially if g^{12} becomes nonnegligible (i.e., in regions of high curvature and rotation).

b. Orthogonal curvilinear coordinates

The above general curvilinear coordinate equations (A1) reduce to orthogonal curvilinear coordinates when $g_{12} = g_{21} = 0$. In this case,

$$G = \sqrt{g_{11}g_{22}}, \qquad g^{11} = \frac{1}{g_{11}}, \qquad g^{22} = \frac{1}{g_{22}},$$

$$u_1 = g_{11}u^1, \qquad u_2 = g_{22}u^2.$$

The physical flow (u, v) is given by

$$u = \frac{1}{\sqrt{g_{11}}}u_1 = \sqrt{g_{11}}u^1 \quad \text{and}$$

$$v = \frac{1}{\sqrt{g_{22}}}u_2 = \sqrt{g_{22}}u^2$$

so that the kinetic energy simplifies as $(1/2)u_ku^k = (1/2)(u^2 + v^2)$. The shallow-water equations (A1) written in terms of the physical flow $\mathbf{u} = (u, v)$ in orthogonal curvilinear coordinates reduce to

$$\partial_t\mathbf{u} + (f + \zeta)\hat{\mathbf{k}} \wedge \mathbf{u} + \boldsymbol{\nabla}\left(\Phi + \frac{1}{2}\mathbf{u}\cdot\mathbf{u}\right) = \mathcal{F}, \tag{A3a}$$

$$\partial_t\Phi + \boldsymbol{\nabla}\cdot\Phi\mathbf{u} = 0. \tag{A3b}$$

REFERENCES

Adcroft, A., C. Hill, and J. Marshall, 1997: Representation of topography by shaved cells in a height coordinate ocean model. *Mon. Wea. Rev.,* **125,** 2293–2315.

Fletcher, C. A. J., 1991: *Computational Techniques for Fluid Dynamics.* 2d ed. Springer-Verlag, 439 pp.

Griffies, S., 2004: *Fundamentals of Ocean Climate Models.* Princeton University Press, 518 pp.

Held, I. M., and M. J. Saurez, 1994: A proposal for the intercomparison of the dynamical cores of atmospheric general circulation models. *Bull. Amer. Meteor. Soc.,* **75,** 1825–1830.

Hundsdorfer, W., and R. Trompert, 1994: Method of lines and direct discretisation: A comparison for linear advection. *Numer. Math.,* **13,** 469–490.

——, B. Koren, M. van Loon, and J. Verwer, 1995: A positive finite-difference advection scheme. *J. Comput. Phys.,* **117,** 35–46.

Jiang, S., P. Stone, and P. Malanotte-Rizzoli, 1999: An assessment of the geophysical fluid dynamics laboratory ocean model with coarse resolution: Annual-mean climatology. *J. Geophys. Res.,* **104,** 25 623–25 645.

Levitus, S., and T. Boyer, 1994: *Temperature.* Vol. 4, *World Ocean Atlas*, NOAA Atlas NESDIS 4, 117 pp.

Marshall, J., A. Adcroft, C. Hill, L. Perelman, and C. Heisey, 1997a: A finite-volume, incompressible Navier-Stokes model for studies of the ocean on parallel computers. *J. Geophys. Res.,* **102,** 5753–5766.

——, C. Hill, L. Perelman, and A. Adcroft, 1997b: Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling. *J. Geophys. Res.,* **102,** 5733–5752.

McGregor, J. K., 1997: Semi-lagrangian advection on a cubic gnomonic projection of the sphere. *Numerical Methods in Atmospheric and Oceanic Modelling: The André J. Robert Memorial Volume,* R. L. Charles, A. Lin, and H. Ritchie, Eds., CMOS/NRC Research Press, 153–169.

McGregor, J. L., 1996: Semi-Lagrangian advection on conformal-cubic grids. *Mon. Wea. Rev.,* **124,** 1311–1322.

Pietrzak, J., 1998: The use of TVD limiters for forward-in-time upstream-biased advection schemes in ocean modeling. *Mon. Wea. Rev.,* **126,** 812–830.

Purser, J., and M. Rancic, 1998: Smooth quasi-homogeneous gridding of the sphere. *Quart. J. Roy. Meteor. Soc.,* **124,** 637–647.

Rancic, M., R. J. Purser, and F. Mesinger, 1996: A global shallow-water model using an expanded spherical cube: Gnomonic versus conformal coordinates. *Quart. J. Roy. Meteor. Soc.,* **122,** 959–982.

Ringler, T., and D. Randall, 2002: Potential enstrophy and energy conserving numerical scheme for solution of the shallow-water equations on a geodesic grid. *Mon. Wea. Rev.,* **130,** 1397–1410.

Ronchi, C., R. Iacono, and P. Paolucci, 1996: The ''cubed sphere.'' A new method for the solution of partial differential equations in spherical geometry. *J. Comput. Phys.,* **124,** 93–114.

Sadourny, R., 1972: Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Mon. Wea. Rev.,* **100,** 136–144.

——, 1975: The dynamics of finite-difference models of the shallow-water equations. *J. Atmos. Sci.,* **32,** 680–689.

Strang, G., 1968: On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.,* **5,** 506–517.

Trenberth, K., J. Olson, and W. Large, 1989: A global ocean wind stress climatology based on ECMWF analyses. NCAR Tech. Rep. NCAR/TN-338+STR, 93 pp.

Williamson, D., J. Drake, J. Hack, R. Jakob, and P. Swarztrauber, 1992: A standard test set for numerical approximations to the shallow-water equations in spherical geometry. *J. Comput. Phys.,* **102,** 211–224.