Geometrically Exact Conservative Remapping (GECoRe): Regular Latitude–Longitude and Cubed-Sphere Grids

PAUL A. ULLRICH

Department of Atmospheric, Oceanic, and Space Sciences, University of Michigan, Ann Arbor, Michigan

PETER H. LAURITZEN

Climate and Global Dynamics Division, National Center for Atmospheric Research, Boulder, Colorado*

CHRISTIANE JABLONOWSKI

Department of Atmospheric, Oceanic, and Space Sciences, University of Michigan, Ann Arbor, Michigan

(Manuscript received 30 September 2008, in final form 1 December 2008)

ABSTRACT

Land, ocean, and atmospheric models are often implemented on different spherical grids. As a conseqence coupling these model components requires state variables and fluxes to be regridded. For some variables, such as fluxes, it is paramount that the regridding algorithm is conservative (so that energy and water budget balances are maintained) and monotone (to prevent unphysical values). For global applications the cubed-sphere grids are gaining popularity in the atmospheric community whereas, for example, the land modeling groups are mostly using the regular latitude–longitude grid. Most existing regridding schemes fail to take advantage of geometrical symmetries between these grids and hence accuracy of the calculations can be lost. Hence, a new Geometrically Exact Conservative Remapping (GECoRe) scheme with a monotone option is proposed for remapping between regular latitude–longitude and gnomonic cubed-sphere grids. GECoRe is compared with existing remapping schemes published in the meteorological literature. It is concluded here that the geometrically exact scheme significantly improves the accuracy of the resulting remapping in idealized test cases.

1. Introduction

Land, ocean, and atmosphere components of coupled climate system models are often implemented on different spherical grids, individually designed to enhance the accuracy or capture features unique to their respective settings. Historically, the regular latitude–longitude (RLL; see Table 1 for a complete list of acronyms used in this paper) grid has been the predominant choice for global atmospheric models, but problems associated with the polar singularity persist, and hence this grid is not well suited for highly scalable atmospheric models. Much interest in recent years has been instead directed toward the development of atmospheric solvers defined on more isotropic spherical grids; for example, the so-called cubed-sphere grid, which divides the polar singularities among eight weaker singularities located at the corners of a cube, and is otherwise highly scalable on parallel architectures. The cubed-sphere grid was originally introduced by Sadourny (1972), and more recently reintroduced by Ronchi et al. (1996) and Rančić et al. (1996) with equiangular grid spacing and orthogonality. For the land component, however, the RLL grid does not pose polar singularity problems as is the case for the atmosphere (with the current complexity of land models). Neither does the land model seem to be susceptible to scalability problems since most of the computation is in vertical columns rather than in the horizontal. Hence, for the foreseeable future the RLL grid

* The National Center for Atmospheric Research is sponsored by the National Science Foundation.

Corresponding author address: Paul Aaron Ullrich, Department of Atmospheric, Oceanic, and Space Sciences, Space Research Building, University of Michigan, 2455 Hayward St., Ann Arbor, MI 48109.
E-mail: paullric@umich.edu

DOI: 10.1175/2008MWR2817.1

TABLE 1. List of acronyms used in this paper.

Acronym	
GECoRe	Geometrically Exact Conservative Remapping
( )-M	suffix stands for monotone filter applied
PCoM	Piecewise constant method
PLM	Piecewise linear method
PPM	Piecewise parabolic method
RLL	Regular latitude–longitude
ABP	Alpha-beta-panel (equiangular cubed-sphere coordinates)
SCRIP	Spherical Coordinate Remapping and Interpolation Package
CaRS	Cascade Remapping between Spherical Grids

seems to be a viable and convenient grid for land model components.

An intricate problem introduced by defining the model components on different spherical grids is that the exchange of information between the grids is nontrivial and requires a regridding algorithm. In a coupled climate system model it is paramount that the regridding process is not a spurious source or sink for first-order moment variables such as mass. To prevent the generation of unphysical negative and/or large values, the regridding must also be shape preserving–monotone for mixing-ratio-related variables. Regridding with these constraints, conservation and monotonicity, is a nontrivial problem if higher than first-order accuracy is desired.

The regridding problem is not only limited to a static grid-to-grid information transfer setting. The problem is essentially the same for finite-volume advection schemes where the mass transport into a given cell is given in terms of integrals over overlapping areas. In fact, methods developed for advection schemes can be readily applied in grid-to-grid regridding problems such as articulated by Margolin and Shashkov (2003). A major difference between the advection regridding problem and static grid-to-grid regridding is that the source or target grid is not static for advection problems. Hence, the regridding algorithm must be able to deal with a large class of source or target grids changing dynamically at each time step. For grid-to-grid regridding the problem is static, facilitating certain parts of the algorithm. For example, the regridding problem can be optimized for specific grid pairs. On the other hand, the advection problem is usually constrained by Courant numbers and the number of source and target grid cells is identical, which constrains the overlap regions. On the contrary, grid-to-grid regridding does not have that constraint so many source grid cells can overlap a particular target grid cell and vice versa.

A strategy for doing conservative regridding without ad hoc conservation fixers is to reconstruct a subgrid-cell distribution in each source grid cell with conservation as a constraint and then integrate the subgrid-cell distributions for the respective source grid cells over the overlap areas. This process of conservative transfer of variables between grids is referred to as remapping or rezoning. Depending on the source and target grid cell geometries (i.e., the overlap regions over which one must integrate) can be very complex. Hence, direct integration on the sphere of the overlap areas seems like an almost impossible task in terms of algorithmic complexity. Note, however, that it has been done in Cartesian geometry in the context of advection (see Rančić 1992). The problem can be greatly simplified by making use of the powerful mathematical theorem, Gauss's divergence theorem, which converts area integrals into line integrals (see Dukowicz and Kodis 1987). This is the approach taken in the most widely used regridding software in the climate community called the Spherical Coordinate Remapping and Interpolation Package (SCRIP; Jones 1999). Also in the algorithm presented in this paper we make use of Gauss's divergence theorem.

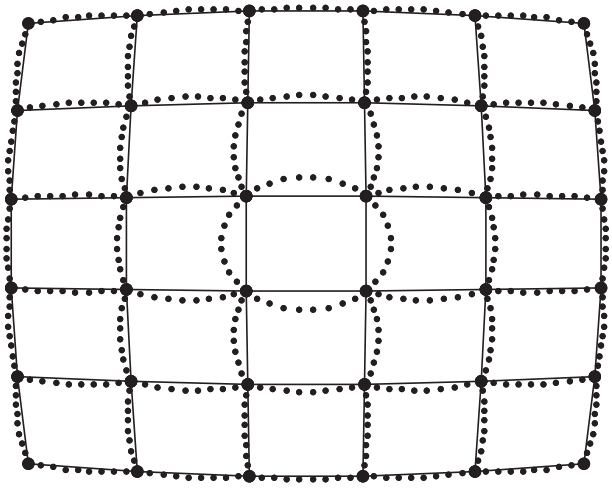

FIG. 1. The cell boundaries of the cubed-sphere south polar panel plotted in Cartesian coordinates. SCRIP approximates cell edges by connecting the cell vertices (filled circles) with straight lines in RLL coordinates (dotted lines). The solid lines are the exact ABP cell walls that are great circle arcs.

To perform the line integrals on the sphere one usually makes simplifying assumptions about the cell sides. For example, the sides of the grid cells are approximated by straight lines in (λ, θ) space in SCRIP. This obviously leads to exact cell wall representations for the RLL grid but other spherical grids such as the cubed-sphere grids do not share that property (see Fig. 1). The remapping algorithm's inability to represent the cell sides exactly is here referred to as the *geometric error* (Lauritzen and Nair 2008, hereafter LN08). In other words, the geometric error is the deviation from exact

closed-form integration along the cell walls. Unfortunately, exact closed-form integration is not always achievable, except for a small set of source and target grids. As a consequence, generic conservative remapping schemes, such as SCRIP and the Cascade Remapping between Spherical Grids (CaRS) scheme, must apply potentially crude geometrical approximations in order to yield a conservative and computationally efficient scheme (Fig. 1). Perhaps serendipitously, we find that complete removal of geometric error via exact closed-form integration is possible for the RLL grid and gnomonic cubed-sphere grids, and hence these grids immediately lend to the development of a geometrically exact conservative scheme.

Assuming small geometric error, the order of the remapping algorithm is determined by the accuracy of the subgrid-cell reconstruction. The errors introduced by the reconstruction are referred to as the *derivative error*. The simplest reconstruction is a piecewise constant (first order) representation in each source grid cell. This method is inherently monotone, but is excessively damping at least in idealized remapping problems. First-order reconstructions are, for example, used in the National Center for Atmospheric Research (NCAR) Coupled Climate System Model, version 3 (CCSM3; Collins et al. 2006), through SCRIP for remapping variables requiring conservation. The effect on climate by using higher-order remapping in a coupled climate system model is unknown (as far as the authors are aware), but it seems instinctive to speculate that it could have a significant effect; especially in areas where remapping is done from a coarse to a fine grid.

Unfortunately, high-order reconstructions are not inherently monotone, making it harder to achieve conservation, monotonicity, and high-order accuracy simultaneously. LN08 choose to apply the cascade remapping approach to achieve monotonocity with high-order reconstructions. This algorithm is referred to as CaRS (i.e., monotone and conservative cascade remapping between spherical grids). In CaRS the remapping problem is split into two one-dimensional problems and hence only one-dimensional limiters are needed to guarantee monotonicity, making it relatively easy to impose shape-preservation simultaneously with higher-order reconstructions. Unfortunately, the CaRS method is inherently prone to geometric errors, although it seems to compensate for these via the higher-order reconstructions that are easily and efficiently applied in one dimension. The geometric errors can be reduced by artificially increasing the resolution in areas of higher geometric error (see LN08 for details). A necessary, but insufficent, condition for completely eliminating the geometric error is to approximate the grid cells in a two-dimensional manner rather than the dimensional split approach such as done in CaRS. However, a fully two-dimensional approach requires fully two-dimensional integrals, reconstructions, and limiters. This greatly increases the complexity of the problem, especially when aiming for higher-order remapping. SCRIP has the option of performing second-order remapping if the user supplies the gradient in latitude–longitude coordinates. In theory SCRIP could be extended to higher-order by including the curvature and high-order derivatives in the line-integral computations but that route was not explored by Jones (1999) and is not a trivial extension.

In this paper we present a new conservative remapping method between gnomonic cubed-sphere grids and the RLL grids hereafter referred to as the Geometrically Exact Conservative Remapping (GECoRe) scheme. GECoRe uses Gauss's divergence theorem to convert area integrals into line integrals. The line integrals are exact (to machine precision) for given polynomial subgrid-cell reconstructions since the lines along which the line integrals are computed exactly coincide with grid lines on the cubed sphere and RLL grid. Consequently the geometric error is completely eliminated in GECoRe. We also use up to third-order-accurate reconstructions and apply limiters to obtain monotonicity thereby obtaining high-order accuracy and shape preservation simultaneously in two dimensions.

The paper is organized as follows. In section 2 we introduce the mathematical basis for GECoRe, that is, how to compute the potentials needed to convert area integrals into line integrals, how line integrals are computed exactly, and how the fully two-dimensional reconstructions are approximated. Section 3 covers some practical considerations such as finding intersections between line segments of the two grids. The accuracy of GECoRe is assessed in section 4 by comparing standard error measures for remapping analytical functions with GECoRe, SCRIP, and CaRS between equi-angular cubed-sphere and RLL grids with different resolutions. A summary is given in section 5.

2. Geometrically Exact Conservative Remapping

In the context of finite-volume methods, we are given the cell-averaged value of a scalar field for each cell in the source grid, denoted by $\overline{f}_n$. The remapping problem then reduces to finding corresponding cell-averaged values in the target grid, denoted $\overline{f}_k$, that accurately represent the underlying scalar field. Throughout this paper we will use the subscript n to denote a quantity corresponding to a cell of the source grid and the subscript k to denote a quantity corresponding to a cell of the target grid.

a. Source and target grids

Of the numerous choices for the cubed sphere, we will focus on the equiangular cubed sphere, where a sphere is decomposed into six identical regions (panels or faces) and the grid lines on each panel are defined by equispaced central angles (see Fig. 2). Hereafter, we will refer to the equiangular cubed sphere as the alpha, beta, panel (ABP) coordinate system, in reference to coordinates on the sphere being identified by the three-element vector (α, β, n_p), where $(\alpha, \beta) \in [-\pi/4, \pi/4]$ are the coordinates on each panel and $n_p \in \{1, 2, 3, 4, 5, 6\}$ is the panel index.

Although we focus on the equiangular projection, we should emphasize that the GECoRe method is trivially extended to any cubed-sphere grid that is composed of grid lines parallel to the panel edges (i.e., gnomonic cubed-sphere grids). Notably, conformal or spring-dynamics grids, where grid lines are ''warped'' near the corners, do not fall into this category [see, e.g., Putman and Lin (2007) for a review of several types of cubed-sphere grids]. Grids with grid lines that are not parallel to panel sides can also be captured via this scheme by replacing the exact line-integral formulas with Gaussian quadrature approximations to the integrals in gnomonic coordinates. Such an approach is taken in the tracer transport scheme described in Lauritzen et al. (2009, manuscript submitted to *J. Comput. Phys.*). Of course, the geometric error will not be completely eliminated if the cell sides are not great spherical arcs. Note that in such a situation the geometric error could be reduced by approximating the cell sides by several great-spherical arc segments.

We will also use so-called gnomonic coordinates interchangeably with equiangular coordinates. Gnomonic coordinates (x, y) are defined in terms of equiangular coordinates via

$$x = a\tan\alpha, \quad y = a\tan\beta, \quad (\alpha, \beta) \in \left[-\frac{\pi}{4}, \frac{\pi}{4}\right]^2, \qquad (1)$$

where a is a constant denoting the edge length of a concentric cube, which, without loss of generality we will take to be equal to unity. The gnomonic projection is important for our analysis since any straight line in the gnomonic projection corresponds to a *spherical arc* on the surface of the sphere.

b. Overview of the method

Typically, a conservative remapping scheme is one that satisfies the global conservation condition:

$$\int_A f_{\text{target}}\, dA = \int_A f_{\text{source}}\, dA, \qquad (2)$$

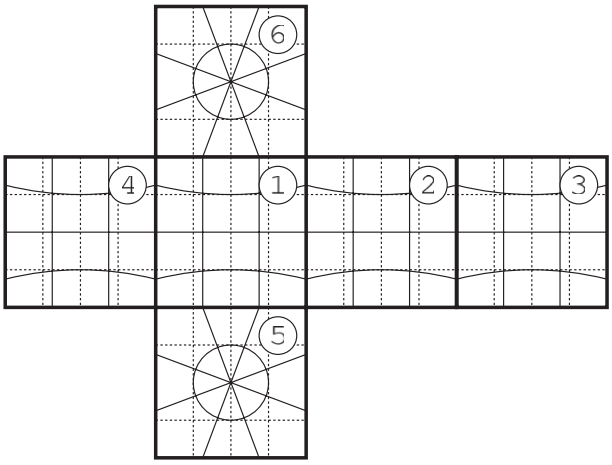

FIG. 2. An illustration of the RLL grid (thin solid lines) and cubed-sphere grid (dotted lines). Thick lines mark the boundaries of each, distinguished by the index given in the upper-right corner. By convention we choose for the southern and northern polar panels to have indices 5 and 6, respectively.

where f_{target} and f_{source} are the global scalar field on the target grid and source grid, respectively. Here the integral is taken over the entire grid surface A (in our case, the surface of a sphere). The stricter local conservation condition states that for every cell k on the target grid, the scalar field must satisfy

$$\overline{f}_k = \frac{1}{A_k}\int_{A_k} f\, dA, \qquad (3)$$

where $\overline{f}_k$ denotes the area-averaged scalar field, f is the global piecewise reconstruction on the source grid, and A_k is the area of cell k. Note that the local conservation condition can be trivially demonstrated to be a sufficient condition for the global conservation condition. Now, if cell k in the target grid overlaps N cells in the source grid, one can write (3) as

$$\overline{f}_k = \frac{1}{A_k}\sum_{n=1}^{N}\int_{A_{n,k}} f_n\, dA, \qquad (4)$$

[see Jones 1999, their Eq. (3)], where $A_{n,k}$ is the area of the source grid cell n that is overlapped by the destination cell k, and f_n is the local value of the scalar field in grid cell n (see Fig. 3). That is, the averaged value in the destination cell is equal to the area-normalized contribution from all overlapping cells on the source grid.

A GECoRe-type remapping scheme can generally be obtained at any order, with the order of the method generally depending on the order of the subgrid-scale reconstruction within each source volume. In general, for a remapping scheme of order h, the subgrid-scale reconstruction takes the form

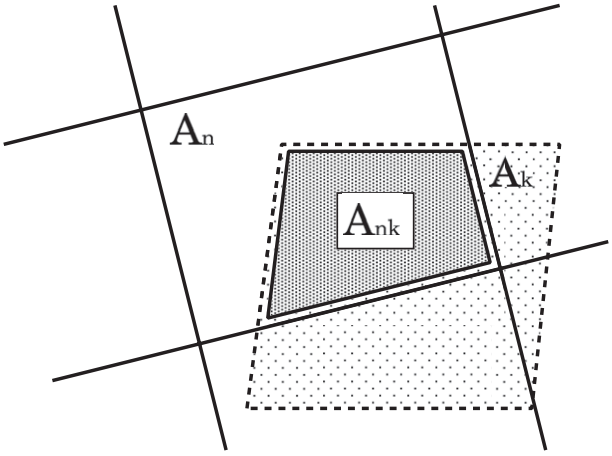

FIG. 3. An example of a quadrilateral target grid cell A_k that overlaps several source grid cells. The region overlapped by both A_k and A_n is denoted by $A_{n,k}$.

$$f_n(x, y) = \sum_{p+q<h} a_n^{(p,q)} (x - x_0)^p (y - y_0)^q, \tag{5}$$

where the reconstruction coefficients $a_n^{(p,q)}$ are constants and x_0 and y_0 denote the x and y components, respectively, of the source-grid cell centroids, defined by

$$x_0 = \frac{1}{A_n} \int_{A_n} x \, dA \quad \text{and} \quad y_0 = \frac{1}{A_n} \int_{A_n} y \, dA. \tag{6}$$

In practice, these quantities are computed by transforming the area integrals to line integrals via Gauss's divergence theorem (as discussed later). To obtain the desired order of accuracy of the method, the reconstruction coefficients must also be obtained via a suitably accurate method. For reasons of consistency, the reconstruction in (5) must also yield the cell-averaged value of the source volume when integrated over the entire source volume, that is,

$$\bar{f}_n = \sum_{p+q<h} a_n^{(p,q)} \frac{1}{A_n} \int_{A_n} (x - x_0)^p (y - y_0)^q \, dA. \tag{7}$$

In appendix A, we present reasonable choices of the reconstruction coefficients that lead to first-, second- and third-order methods.

The conservative remapping scheme that follows from (4) and (5) can then be written as

$$\bar{f}_k = \sum_{n=1}^{N} \sum_{p+q<h} a_n^{(p,q)} w_{n,k}^{(p,q)}, \tag{8}$$

where, following Jones (1999), we have defined the mesh-dependent weights via

$$w_{n,k}^{(p,q)} = \frac{1}{A_k} \int_{A_{n,k}} (x - x_0)^p (y - y_0)^q \, dA. \tag{9}$$

The form in (8) is particularly meaningful, as it emphasizes the separation of the purely reconstruction-dependent coefficients $a_n^{(p,q)}$ and the purely mesh-dependent subcell weights $w_n^{(p,q)}$.

Following Dukowicz and Kodis (1987), we compute the weights in (9) by converting them into line integrals using the divergence theorem. Consider an arbitrary vector field $\boldsymbol{\Psi}$ defined in terms of ABP unit basis vectors $(\mathbf{e}_\alpha, \mathbf{e}_\beta)$ as $\boldsymbol{\Psi} = \Psi_\alpha \mathbf{e}_\alpha + \Psi_\beta \mathbf{e}_\beta$. In general 2D curvilinear coordinates the divergence is given by

$$\nabla \cdot \boldsymbol{\Psi} = \frac{1}{\sqrt{g}} \left[\frac{\partial}{\partial x^1} (\sqrt{g} \Psi^1) + \frac{\partial}{\partial x^2} (\sqrt{g} \Psi^2) \right], \tag{10}$$

where Ψ^1 and Ψ^2 are the contravariant components of the vector $\boldsymbol{\Psi}$ and g is the determinant of the metric. Hence, specifically for cubed-sphere [see (D8) and (D9)] we have

$$\nabla \cdot \boldsymbol{\Psi} = (\rho^3 \cos^2\alpha \cos^2\beta) \left[\frac{\partial}{\partial \alpha} \left(\frac{\Psi_\alpha}{\rho \cos\beta} \right) + \frac{\partial}{\partial \beta} \left(\frac{\Psi_\beta}{\rho \cos\alpha} \right) \right], \tag{11}$$

where

$$\rho = \sqrt{1 + \tan^2\alpha + \tan^2\beta}. \tag{12}$$

After integrating (11) over the subcell and assuming sufficiently smooth boundaries, we can write the resulting expression in the form:

$$\int_{A_{n,k}} \nabla \cdot \boldsymbol{\Psi} \, dV = \int_{\beta_1}^{\beta_2} \int_{\alpha_1(\beta)}^{\alpha_2(\beta)} \frac{\partial}{\partial \alpha} \left(\frac{\Psi_\alpha}{\rho \cos\beta} \right) d\alpha \, d\beta + \int_{\alpha_1}^{\alpha_2} \int_{\beta_1(\alpha)}^{\beta_2(\alpha)} \frac{\partial}{\partial \beta} \left(\frac{\Psi_\beta}{\rho \cos\alpha} \right) d\beta \, d\alpha, \tag{13}$$

where α_1, α_2, $\beta_1(\alpha)$, and $\beta_2(\alpha)$ represent the boundaries of the domain of integration. Then, on applying the fundamental theorem of calculus, we obtain the divergence theorem for cubed-sphere coordinates,

$$\int_{A_{n,k}} \nabla \cdot \boldsymbol{\Psi} \, dV = - \oint_{\partial A_{n,k}} \left(\frac{\Psi_\alpha}{\rho \cos\beta} \, d\beta + \frac{\Psi_\beta}{\rho \cos\alpha} \, d\alpha \right), \tag{14}$$

where the contour integral is taken in the counterclockwise direction around the boundary of a given overlapping volume $A_{n,k}$, here denoted by $\partial A_{n,k}$. Note that the Jacobian term obtained by expanding the area integral in (13) cancels with the $\sqrt{g}$ from the divergence in (10) and so does not appear in the final form of the contour integral. The spatial curvature instead comes into play when solving for $\boldsymbol{\Psi}$ via (10).

To apply the divergence theorem to compute the weights associated with each line segment over an integrable scalar field $\phi(\alpha, \beta, n_p)$, we must first obtain a potential $\boldsymbol{\Psi}$ associated with that field—namely, one that satisfies

$$\nabla \cdot \boldsymbol{\Psi} = \phi. \tag{15}$$

Note that for a given ϕ, there does not exist a unique potential $\boldsymbol{\Psi}$, but a family of potentials $\boldsymbol{\Psi}$ that satisfy (15). However, one can obtain a unique potential on imposing $\Psi_\alpha = 0$ and choosing any constant of integration for the potential to equal zero. This limitation is largely by convention, as we could also choose $\Psi_\beta = 0$, for example. However, we find that integration on the RLL grid is generally easier upon imposing the former constraint, and so we will henceforth apply $\Psi_\alpha = 0$ when deriving each potential. Furthermore, we restrict our attention to solving integrals of the form in (9)—noting that the integrand of (9) can be expanded as terms of the form $x^p y^q$ [using (1)]—and so use the notation $\boldsymbol{\Psi}^{(p,q)}$ to denote the potential associated with the scalar field $\tan^p\alpha \tan^q\beta$. That is, we use (11) to define $\boldsymbol{\Psi}^{(p,q)}$ via the differential equation:

$$\frac{\partial}{\partial\beta}\left(\frac{\Psi_\beta^{(p,q)}}{\rho\cos\alpha}\right) = \frac{\tan^p\alpha \tan^q\beta}{\rho^3\cos^2\alpha\cos^2\beta}, \quad \text{with } \Psi_\alpha^{(p,q)} = 0. \tag{16}$$

Note that one may solve (16) in terms of either equiangular or gnomonic coordinates, which are connected via the relation in (1). In either case we will obtain an identical expression for the potential.

Since our search algorithm will provide a list of line segments, rather than a list of contours, a computational implementation of (14) will take the following form:

$$\int_{A_{n,k}} \tan^p\alpha \tan^q\beta \, dA = -\sum_s I_s^{(p,q)}\Big|_{(\partial A_{n,k})s}, \tag{17}$$

where the summation is taken over all line segments s along the boundary of subcell $A_{n,k}$. Here, $I_s^{(p,q)}$ is shorthand notation for the antiderivative over the potential field obtained from (16),

$$I_s^{(p,q)} = \int \frac{\Psi_\beta^{(p,q)}}{\rho\cos\alpha}\, d\alpha, \tag{18}$$

and hence it is evaluated at the endpoints of each line segment. A detailed presentation of the calculations and implementation details required for the first-, second-, and third-order-accurate schemes are given in appendix A.

c. Summary of the GECoRe algorithm

A GECoRe remapping scheme is initialized as follows:

1) Perform a search on the source and target grids, classifying line segments by type (i.e., constant α, constant β, constant latitude, or constant longitude) and keeping track of their endpoints and orientation. Each line segment should then be associated with at most one ABP cell and one RLL cell.
2) Using (9) calculate the weights $w_{n,k}^{(p,q)}$ associated with each line segment over the fields $\tan^p\alpha \tan^q\beta$, for $p + q < h$, where h is the order of the method [this leads to $(1/2)h(h + 1)$ weights per line segment]. The weights of each line segment are computed by simply evaluating the associated antiderivative $I_s^{(p,q)}$ at each endpoint (see appendix A). Note that one can save memory and online computation time by instead storing the sum of all weights for a given overlap cell rather than the weights for individual line segments.

Once the GECoRe scheme is initialized for a particular mesh pair, the resulting initialization data can be saved to a file for later use. The actual remapping is then performed as follows:

1) Calculate the reconstruction coefficients $a_s^{(p,q)}$ associated with the scalar field, potentially using neighboring cell values.
2) Use the weights $w_{n,k}^{(p,q)}$ computed in the initialization step, along with (8) to compute the remapped field in each of the target grid cells.

3. Practical considerations

In this section we present issues relating to the implementation of the GECoRe scheme.

a. Search algorithm

Since we are restricted to RLL and cubed-sphere grids, the search algorithm for finding line segments is dramatically simplified when compared to the SCRIP algorithm. In fact, this knowledge of the coordinate systems allows us to exactly calculate the line segment endpoints up to machine precision. The proposed technique involves first searching along longitude and latitude lines to compute intersection points, binning each line accordingly depending on its corresponding ABP cell. Then a search within each cell can be performed to obtain all lines of constant α and β within an RLL cell. This binning procedure results in memory locality of line segments associated with a given RLL cell, and hence is optimal for remappings from the RLL grid to ABP grid. A similar algorithm can be performed to

obtain line segments binned by ABP cell, or the results from the forward algorithm can simply be resorted to obtain the desired result. Note that special attention must be paid to special cases, such as coincident lines, so as to avoid double counting of line segments.

b. Spherical coordinates

Similar to the SCRIP algorithm, certain aspects of the spherical coordinate system introduce additional problems when applied in practice. Unlike SCRIP, the pole points do not pose any particular problem since all calculations are performed along the surface of the cubed sphere, which has no special treatment of the pole points. However, when calculating line segment weights the multiple-valued longitude coordinate in spherical coordinates must be taken into account. Antiderivatives that require the evaluation of the longitude coordinate at each endpoint must ensure that only the "shortest" distance between longitudes is used. Similarly, additional checks must be performed to ensure that antiderivatives that require the evaluation of an arctangent have endpoints evaluated along the same branch of the arctangent function. Failure to take into account either of these factors will result in spurious factors of π being introduced into the calculation over line segments where the longitude coordinate becomes discontinuous.

c. Extensions to higher orders

Although it is not proven here, symbolic computations have shown that the line segment weights $w_{n,k}^{(p,q)}$ can be computed in exact closed form up to any choice of p and q. However, as mentioned earlier, the number of weights (and reconstruction coefficients) that need to be computed for a method of order h is quadratic in h, and hence the resulting computation becomes increasingly infeasible at higher orders. Furthermore, on increasing the order of the scheme, one finds that the resulting antiderivatives also become increasingly complicated expressions (observe the differences between antiderivatives for the first-, second-, and third- order schemes given in sections a, b, and c, respectively, in appendix A).

d. Parallelization considerations

The nonlocality of this algorithm required during online calculations is largely constrained to calculating the reconstruction coefficients associated with each cell (which requires a stencil size that increases with the order of accuracy of the method). In this sense we conclude that the GECoRe scheme is potentially highly parallelizable, for example by using parallelization techniques currently employed in an existing finite-volume model.

4. Results

The new remapping algorithm (GECoRe) has been implemented for both RLL to cubed sphere and cubed sphere to RLL remapping schemes and tested on a variety of analytical fields. For comparison, we have provided results from SCRIP and CaRS for the piecewise constant first-order reconstruction (PCoM), piecewise linear second-order reconstruction (PLM), and piecewise parabolic third-order reconstruction (PPM). Particular interest should be paid to comparing the results from SCRIP and GECoRe, since much of the underlying structure of the algorithms in these two cases are directly comparable. We note, however, that the method referred to in this paper as SCRIP PPM is an extension of the SCRIP scheme of Jones (1999) to high-order accuracy. Here we have implemented this scheme by using the PPM method of Colella and Woodward (1984) to compute left and right edge values (ϕ_L and ϕ_R, respectively) in each cell, and hence use these values for reconstructing the gradient within each cell. As a consequence, this method is not a true PPM method, in that the subgrid-scale reconstruction is not composed of parabolic terms. As of the current time, no true PPM implementation of SCRIP is believed to exist.

a. Test cases

Our analysis mirrors the approach of LN08, in that we consider three idealized test cases and computed error measures for both equiangular cubed sphere to RLL remapping and vice versa.

Following Jones (1999) and LN08 a relatively smooth function resembling a spherical harmonic of order 2 and azimuthal wavenumber 2 (see Fig. 4a),

$$\psi = 2 + \cos^2\theta \cos(2\lambda), \quad (Y_2^2), \tag{19}$$

and a relatively high frequency wave similar to a spherical harmonic of order 32 and azimuthal wavenumber 16 (see Fig. 4b),

$$\psi = 2 + \sin^{16}(2\theta) \cos(16\lambda), \quad (Y_{32}^{16}), \tag{20}$$

are used. These waves are useful for testing the performance of the algorithm for a large-scale well-resolved field as well as a higher-frequency wave in the midlatitudes with relatively rapidly changing gradients. In addition, as in LN08, we test all three schemes with the dual stationary vortex fields (Nair and Machenhauer 2002), since this test leads to significant variation of the field over the cubed-sphere corners (see Fig. 4c). The analytical form of this field is given by

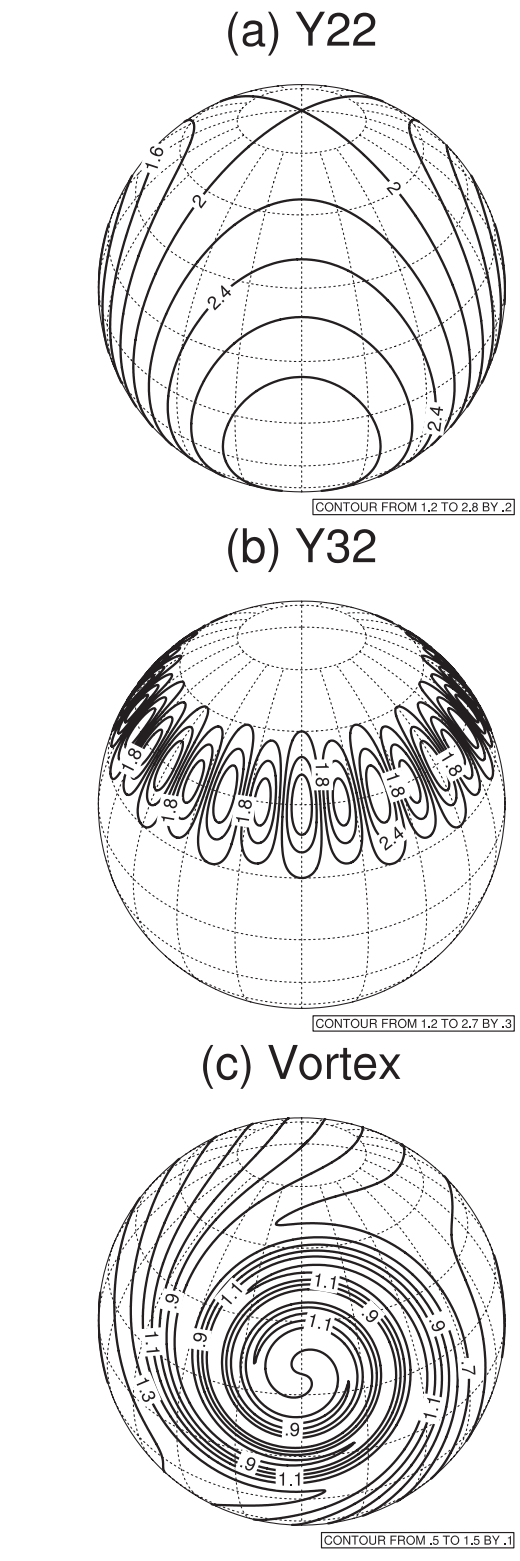

FIG. 4. Contours of the analytical function (a) Y_2^2, (b) Y_{32}^{16}, and (c) the vortex fields with one of the vortices centered about $(\lambda_0, \theta_0) = (0, 0, 6)$. Dotted lines show the regular latitude–longitude grid.

$$\psi = 1 - \tanh\left[\frac{\rho'}{d}\sin(\lambda' - \omega' t)\right], \quad (V_X), \tag{21}$$

where the radius $\rho' = r_0 \cos\theta'$, with angular velocity:

$$\omega'(\theta') = \begin{cases} 0 & \text{if } \rho' = 0, \\ \dfrac{V_t}{\rho'} & \text{if } \rho' \neq 0, \end{cases} \tag{22}$$

and normalized tangential velocity:

$$V_t = \frac{3\sqrt{3}}{2}\operatorname{sech}^2\rho' \tanh\rho'. \tag{23}$$

The (λ', θ') refer to a rotated spherical coordinate system with a pole located at (λ_0, θ_0). Following LN08 we choose $(\lambda_0, \theta_0) = (0, 0, 6)$, $r_0 = 3$, $d = 5$, and $t = 6$.

b. Error measures

The performance of the algorithm is quantified using standard error measures:[1]

$$l_1 \equiv \frac{I(|\overline{\psi}_{\text{num}} - \overline{\psi}_{\text{exact}}|)}{I(\overline{\psi}_{\text{exact}})}, \tag{24}$$

$$l_2 \equiv \frac{\sqrt{I[(\overline{\psi}_{\text{num}} - \overline{\psi}_{\text{exact}})^2]}}{\sqrt{I[(|\overline{\psi}_{\text{exact}}|)^2]}}, \quad \text{and} \tag{25}$$

$$l_\infty \equiv \frac{\max(|\overline{\psi}_{\text{num}} - \overline{\psi}_{\text{exact}}|)}{\max(|\overline{\psi}_{\text{exact}}|)}, \tag{26}$$

where I is the global integral:

$$I(\overline{f}) = \sum_n \overline{f}_n A_n. \tag{27}$$

The numerically generated "exact" solution

$$\overline{f}_n = \frac{1}{A_n}\int_{A_n} f\, dA, \tag{28}$$

is computed by fourth-order Gaussian quadrature. One finds that the error measure l_1 tends to identify the error in large-scale features of the field, whereas l_2 identifies error in small-scale features. The l_∞ measure, on the other hand, identifies the maximum relative cellwise error over the entire field.

c. Calculation of reconstruction coefficients

Calculation of the reconstruction coefficients $a^{(p,q)}$ that describe the subgrid-cell reconstruction is required for the second- and third-order schemes. Recall that the grid is an equiangular cubed-sphere grid that, when

[1] Note that the l_2 error employed in LN08 corresponds to $(l_2)^2$.

translated to gnomonic coordinates, leads to cell centroids that are far from equidistant. As a consequence, if applied in gnomonic coordinates, an equidistant discrete derivative operator will lead to large derivative errors. We examine two possible solutions to this problem: First, we can compute the derivatives in equiangular coordinates using an equidistant discrete derivative operator and then apply a stretching factor to obtain the derivatives in gnomonic coordinates. Second, we can use a nonequidistant discrete derivative operator in gnomonic coordinates. These methods will be compared, in terms of the resulting error measures, in section 4e.

We briefly discuss the method of computing the reconstruction coefficients via the nonequidistant fitting in gnomonic coordinate space. In general, these coefficients can be computed by fitting a parabola through the neighboring centroids, where we have assumed that these centroids take on the cell-averaged value, and extracting the reconstruction coefficients from the quadratic coefficients. If we define

$$\Delta x_L = x_{i-1} - x_i \quad \text{and} \quad \Delta x_R = x_{i+1} - x_i, \tag{29}$$

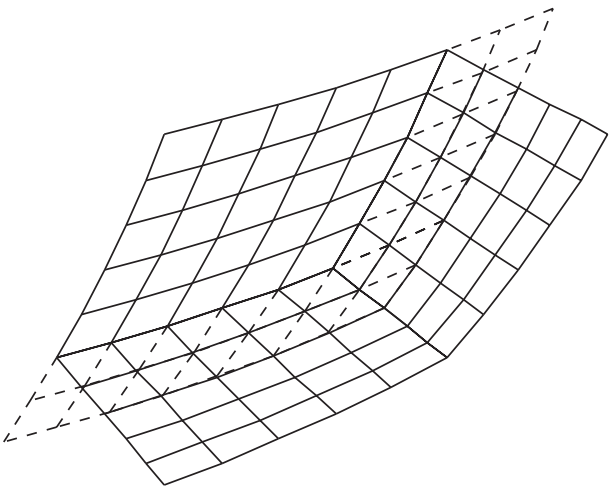

FIG. 5. A depiction of the halo region along the boundary of the top surface (dashed lines), showing the overlap with cells of the neighboring surfaces. Observe that accurate modeling of the halo region only requires 1D interpolation for this choice of grid.

it follows that a parabola $p(x)$ fitted through the points (x_{i-1}, y_{i-1}), (x_i, y_i), and (x_{i+1}, y_{i+1}) will satisfy

$$\left(\frac{\partial p}{\partial x}\right)_i = \frac{(y_{i-1})(\Delta x_R)^2 - (y_i)[(\Delta x_R)^2 - (\Delta x_L)^2] - (y_{i+1})(\Delta x_L)^2}{(\Delta x_R)(\Delta x_L)(\Delta x_R - \Delta x_L)}, \tag{30}$$

and

$$\frac{1}{2}\left(\frac{\partial^2 p}{\partial x^2}\right)_i = \frac{(y_{i-1})(\Delta x_R) - (y_i)(\Delta x_R - \Delta x_L) - (y_{i+1})(\Delta x_L)}{(\Delta x_R)(\Delta x_L)(\Delta x_L - \Delta x_R)}. \tag{31}$$

Note that the discretized derivatives in (30) and (31) reduce to the usual central difference discretization in the case of equispaced grid points ($\Delta x_L = -\Delta x_R$). A discretization for the third-order cross term $(\partial^2 f/\partial x \partial y)_n$ can be easily calculated on repeatedly applying (30) in each coordinate direction. It is simple but mathematically intensive to extend this method to higher orders by fitting a quartic or higher-order curve, and so the resulting formula is not presented here.

Although both the parabolic fit (three point) and quartic fit (five point) can be used to derive reconstruction coefficients for the third-order scheme, we have chosen to use the parabolic fit for the second-order scheme and the quartic fit for the third-order scheme. This choice is made since the parabolic fit corresponds most closely to a piecewise-linear reconstruction, whereas the quartic fit corresponds most closely to the piecewise-parabolic reconstruction of Colella and Woodward (1984).

Note that when applying the discretized derivative operator in ABP coordinates, halo regions must be provided for cells along the boundary of each panel. These halo cells should correspond to the cells that would be obtained by extending each panel outward, overlapping cells of the neighboring panels (see Fig. 5). Hence, halo cells do not exactly correspond to boundary cells on the neighboring panels, and their cell-averaged values must be obtained via a 1D remapping. Note that this 1D remapping does not require that the conservation criteria be fulfilled, since the area-averaged property of the interior cells is satisfied for any choice of reconstruction coefficients. For our purposes, we obtained the best accuracy from a straightforward fourth-order nonconservative cubic fit.

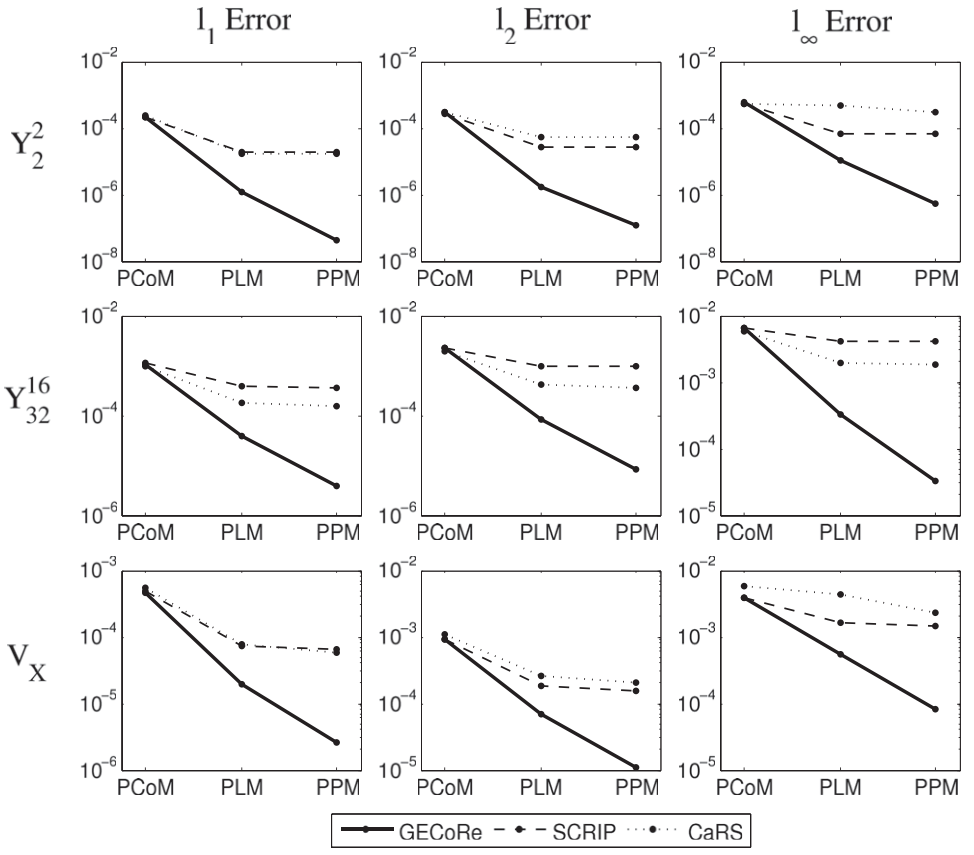

FIG. 6. Performance measures for the remapping of Y_2^2, Y_{32}^{16}, and the idealized vortices (V_X) from a medium-resolution ABP grid ($N_c = 80$) to an RLL grid ($N_\lambda = 128, N_\theta = 64$) using GECoRe, SCRIP, and CaRS with PCoM, PLM, and PPM reconstructions: (left) l_1, (middle) l_2, and (right) l_∞ error.

d. Discussion

The error measures associated with remapping from the equiangular cubed-sphere grid to RLL grid are given in Figs. 6 and 7, for a high-resolution cubed-sphere grid ($N_c = 80$ grid lines on each panel) and a medium-resolution cubed-sphere grid ($N_c = 40$) mapped to a RLL grid with $N_\lambda = 128$ longitudes and $N_\theta = 64$ latitudes. The results of remapping from the same RLL grid ($N_\lambda = 128$, $N_\theta = 64$) to a high-resolution cubed-sphere grid ($N_c = 80$) are given in Fig. 8.

It should be emphasized that, in each case, the GECoRe method should perform *at least as well as* SCRIP, since both methods lead to a derivative error that should be roughly identical for PCoM and PLM. The two methods deviate only in their geometric error, in that the geometrically exact techniques used for GECoRe lead to geometric error that is roughly on the order of machine epsilon. It should be noted that slight deviations from machine epsilon occur in GECoRe because of poorly conditioned function evaluations in some of the antiderivatives (i.e., calculations involving the difference of two nearly equal floating point numbers, or evaluations of arcsine or arccosine near ± 1), but for our purposes we can assume these deviations are effectively negligible.

As expected, all error norms for SCRIP, CaRS, and GECoRe tend to decrease when going from PCoM to PLM to PPM. We expect that extending these methods by including the piecewise cubic scheme (PCM, as done for CaRS in LN08) and higher-order reconstructions will lead to smaller, and perhaps worthwhile, improvements in the accuracy of the method (Figs. 6, 7, and 8).

For the first-order piecewise constant method, we observe nearly identical behavior for GECoRe and SCRIP since, in both cases, line segments are effectively integrated along constant fields. Hence, small perturbations in the geometrical orientation of the line segments will not lead to significant differences in the resulting line integral.

However, we find that the error measures in GECoRe and SCRIP deviate significantly for the second- and third-order methods. The effect of geometric error is clearly apparent in our calculations, as GECoRe produces results that are often one or two orders of magnitude better than

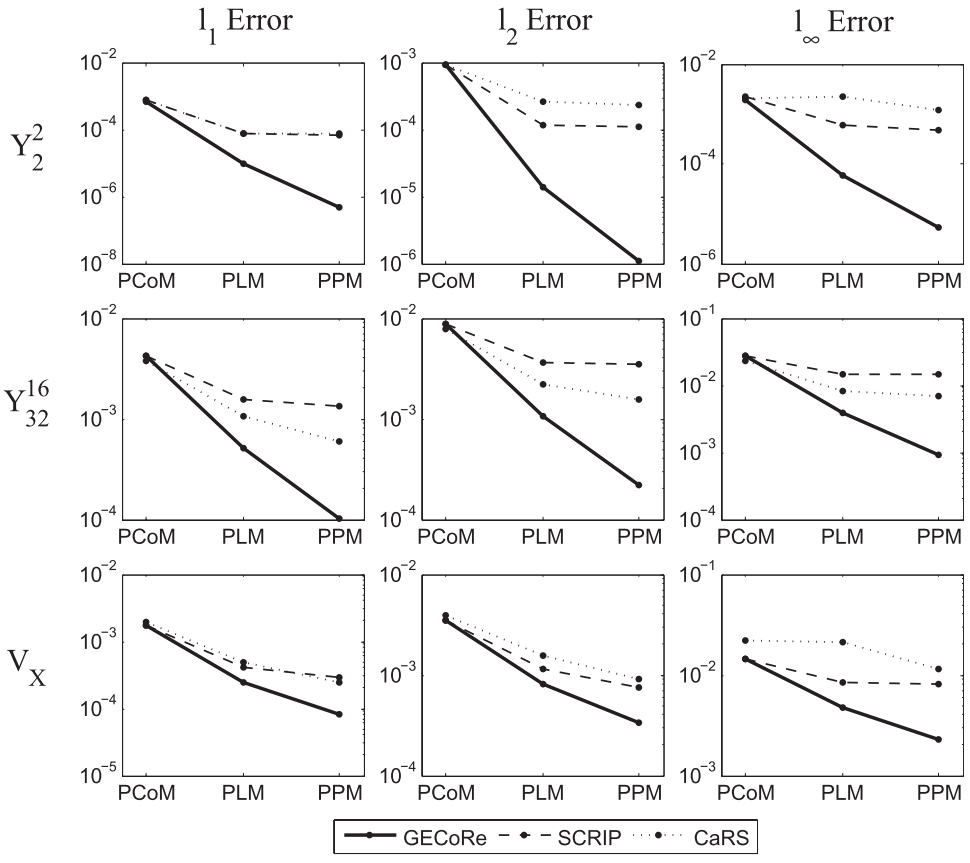

FIG. 7. As in Fig. 6, but with $N_c = 40$.

the associated error from SCRIP. The results are particularly apparent for the smooth field Y_2^2, where each of the error measures shows an improvement of two–four orders of magnitude.

Comparing the spatial distribution of error for the three schemes (see Fig. 9), we observe that the error for SCRIP and CaRS tends to be most significant near the corners, whereas GECoRe has a much more uniform distribution of the resulting error. This result reflects the lack of a contribution from geometric error in the GECoRe scheme, which tends to dominate near the singularities of the cubed-sphere grid. Hence, we can further conclude that, compared to SCRIP and CaRS, the GECoRe scheme tends to be affected less strongly by the shape of the underlying grid.

e. Impact of the reconstruction method

We briefly turn our attention to derivative error in the GECoRe schemes by comparing four methods for computing the reconstruction coefficients in each cell. We focus on the nonequidistant parabolic fit (three point) and quartic fit (five point) methods, both computed in gnomonic coordinates, to the stretched equidistant three- and five-point methods computed in equiangular coordinates. Results from this comparison are given in Fig. 10.

Observe that there is no significant difference between both three-point schemes and both five-point schemes for the nonsmooth Y_{32}^{16} and V_X test cases, since the natural derivative error dwarfs any error that would be present from stretching of the equiangular reconstruction. However, for the smooth test case, there are obvious deviations between the four methods in the third-order PPM scheme. In particular, we clearly observe an immediate benefit to computing the reconstruction coefficients directly in gnomonic coordinates. From the nonsmooth test cases we also observe an obvious benefit to increasing the stencil size.

f. Impact of the monotone filter

To ensure monotonicity in the reconstruction, we employ the monotone filter of Barth and Jespersen (1989). This simple monotone filter simply scales the subgrid-scale reconstruction so that its minimum and maximum values do not exceed the cell averages of the neighboring cells. In the case of the second-order linear reconstruction, the extreme values within a cell will occur at the four corner points. For the third-order reconstruction,

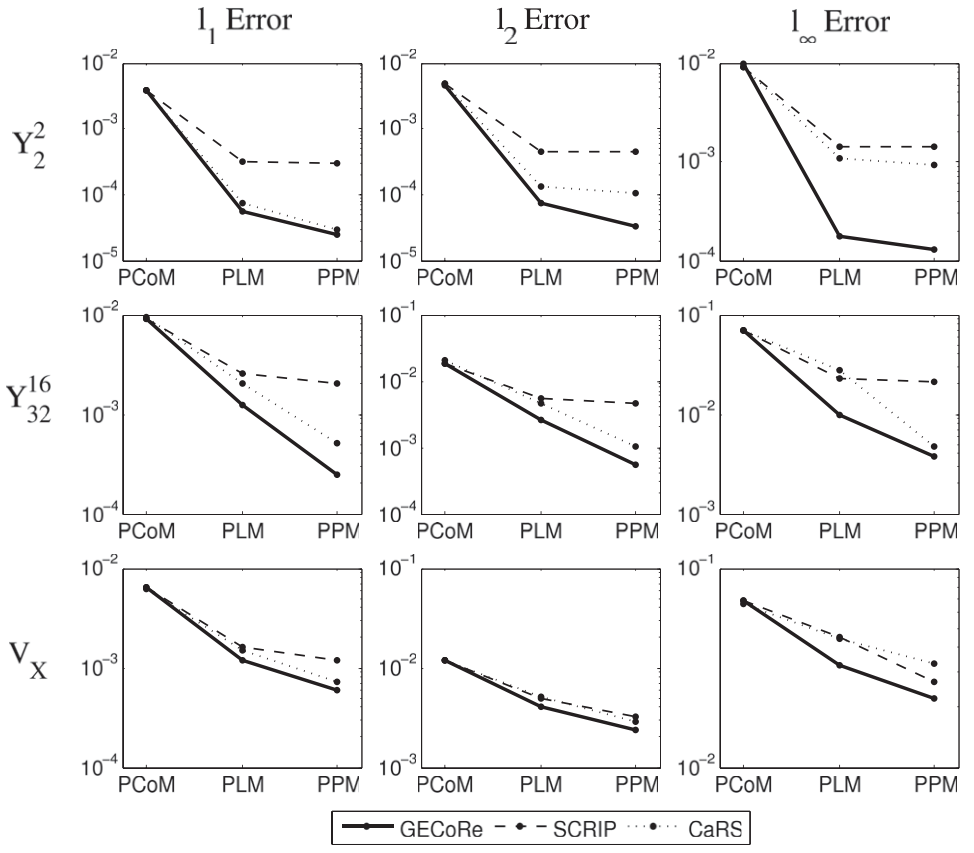

FIG. 8. As in Fig. 6, but remapping from an RLL grid ($N_\lambda = 128$, $N_\theta = 64$) to an ABP grid ($N_c = 80$).

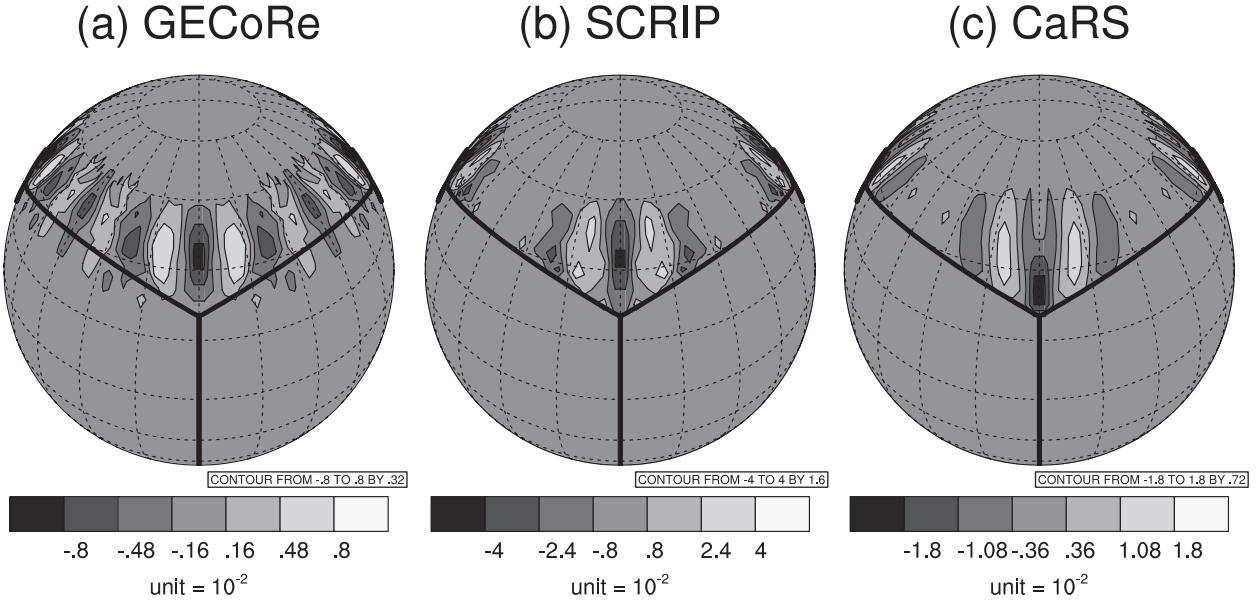

FIG. 9. Spatial distribution of the error ($\times 10^{-2}$) for the remapping of Y_{32}^{16} from a coarse-resolution ABP grid ($N_c = 40$) to an RLL grid ($N_\lambda = 128$, $N_\theta = 64$) using (a) GECoRe, (b) SCRIP, and (c) CaRS with piecewise parabolic (third order) reconstructions.

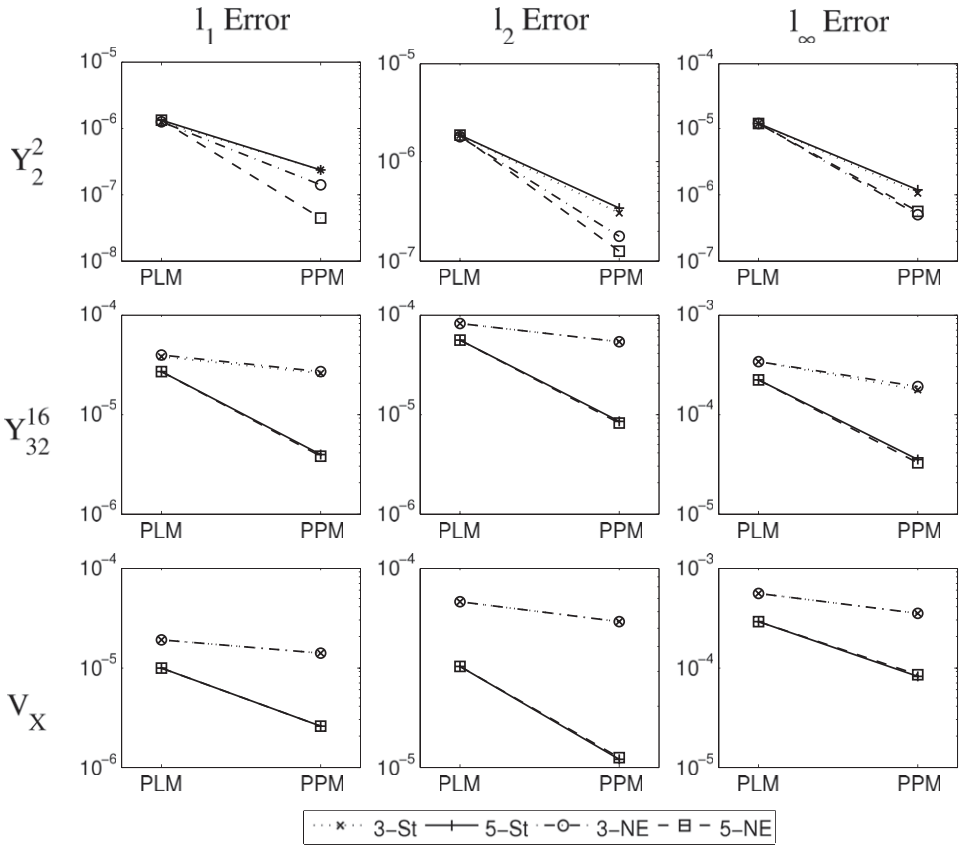

FIG. 10. Performance measures for the remapping of Y_2^2, Y_{32}^{16}, and the idealized vortices (V_X) from a high-resolution ABP grid ($N_c = 80$) to an RLL grid ($N_\lambda = 128, N_\theta = 64$) using GECoRe with four choices of subgrid-scale reconstruction techniques: three-point stretched equiangular (3-St), five-point stretched equiangular (5-St), three-point nonequidistant gnomonic (3-NE), and five-point nonequidistant (5-NE): (left) l_1, (middle) l_2, and (right) l_∞ error.

the extrema could also possibly occur along the boundary or within the cell. Hence, five additional points must be checked in the third-order scheme.

The effect of applying the monotone filter to the remapping scheme is given in Fig. 11. We observe that this simple monotone limiter tends to reduce the accuracy of the method by an order of magnitude, but, as a consequence, maintains that global extreme points are not enhanced. Tests performed on a simple cosine hill field, which is more susceptible to overshoots and undershoots, actually result in improved accuracy of the remapped field under the monotone limiter (not shown).

As resolution is increased, we expect that the loss of accuracy due to the monotone limiter will be reduced, since higher resolution results in less relative variation in the scalar field. Furthermore, advanced limiters, such as that of Zerroukat et al. (2005), if extended to two dimensions, are certain to result in an improved accuracy of these results.

5. Summary

A general modeling environment consists of different model components, usually implemented on different grid systems. Hence, accurate translation of scalar field data between grid systems is important in order to ensure overall accuracy of the model. For variables such as fluxes conservation is a particularly important property that should also be maintained by the remapping scheme, as it ensures that a given model does not disobey the fundamental laws of nature. The accuracy of existing conservative schemes, such as the Spherical Coordinate Remapping and Interpolation Package and the Cascade Remapping between Spherical Grids scheme is often limited by the capacity of these methods to accurately model the geometry of the problem. Here we have presented a high-order geometrically exact scheme for conservative and monotone remapping of scalar fields between the regular latitude–longitude and gnomonic cubed-sphere geometries. The new remapping scheme

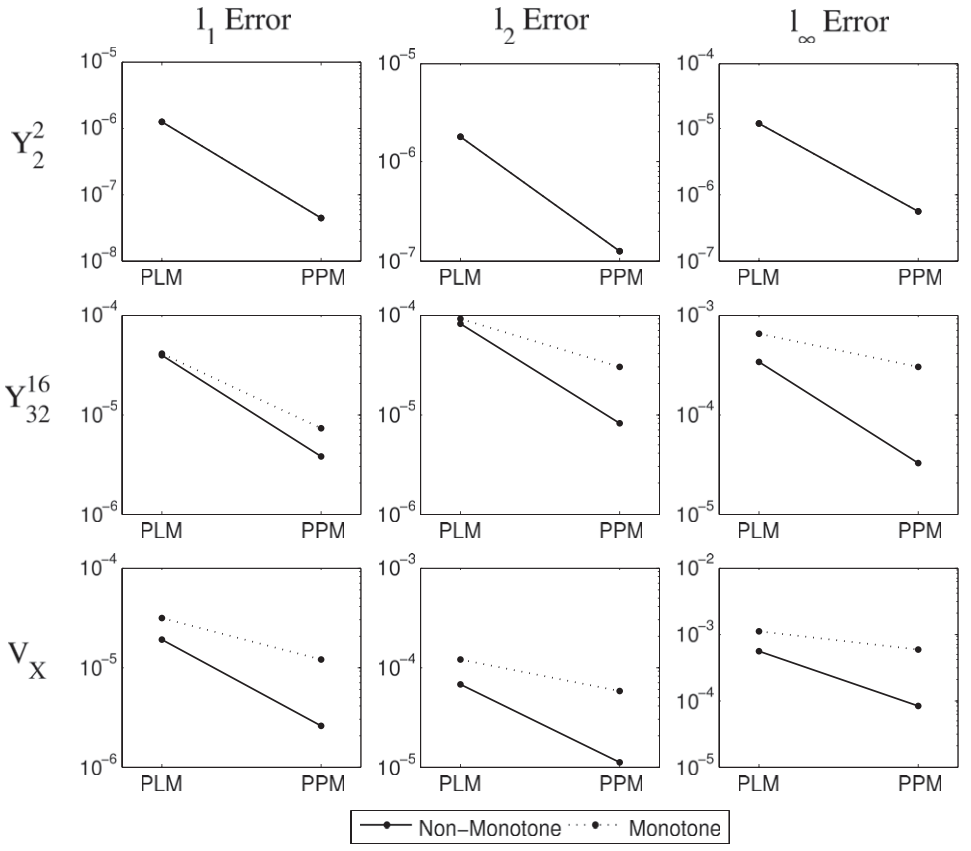

FIG. 11. Performance measures for the remapping of Y_2^2, Y_{32}^{16}, and the idealized vortices (V_X) from a high-resolution ABP grid ($N_c = 80$) to an RLL grid ($N_\lambda = 128, N_\theta = 64$) using GECoRe with (dotted line) and without (solid line) monotone filtering: (left) l_1, (middle) l_2, and (right) l_∞ error.

is referred to as Geometrically Exact Conservative Remapping.

GECoRe is based on the principle that we can integrate certain fields in exact closed-form along grid lines in the gnomonic cubed-sphere grid and regular latitude–longitude grid. The advantage of this approach is the removal of geometric error associated with low-order approximations to line segments at a similar computational cost to existing techniques. Here we provide mechanisms for constructing schemes up to the third-order, possibly combined with an inexpensive monotone filter.

GECoRe is validated by remapping a standard set of both smooth and rapidly varying test functions. Standard error measures are compared with existing SCRIP and CaRS schemes at low and medium resolutions of the cubed-sphere grid. We observe that the GECoRe scheme excels in both cases when applied at second order or higher, often yielding a one or two order of magnitude improvement over the existing schemes without additional tweaking. However, SCRIP is more general than GECoRe, since it can, in principle, handle any kind of spherical grid. Generalized versions of GECoRe can be obtained by replacing the exact integrals, where the exact formulas are too complicated or do not exist, with Gaussian quadrature of the potentials (Lauritzen et al. 2009, manuscript submitted to *J. Comput. Phys.*).

Acknowledgments. The authors thank Dr. Ram Nair for his comments and suggestions during the development of this paper and Dr. Rich Loft for his encouragement and administration of the SIParCS program, which allowed this work to be produced. The first author is grateful to NCAR's SIParCS program and the University of Michigan for providing funding for this project.

APPENDIX A

Calculation of Antiderivatives

In general, for interpolation between the RLL grid and ABP grid, we must evaluate the contour integral in (17) along four types of line segments:

1) Lines of constant α. Since $d\alpha = 0$, any integral along a line of constant α will always evaluate to zero.

Hence lines of constant α can be ignored in the computation.

2) Lines of constant β. In this case the integral can be evaluated directly from (18), on taking β = constant.
3) Lines of constant longitude (λ). In this case we can rewrite the integrand of (18) in terms of α and λ via the RLL and ABP coordinate relations given in (D1) and (D3) and then integrate to obtain a closed-form expression. Alternatively, transforming these integrals to make use of θ as the dummy variable and then rewriting all α and β components in terms of λ and θ may be helpful when performing the integration. Furthermore, observe that lines of constant λ are lines of constant α on panels 1–4, and hence the resulting integral will evaluate to 0.
4) Lines of constant latitude (θ). In this case we can rewrite the integrand of (18) in terms of α and θ via the RLL and ABP coordinate relations given in (D1) and (D3) and then integrate to obtain a closed-form expression. Alternatively, it may be helpful to make use of λ as the dummy variable and rewrite all α and β components in terms of λ and θ prior to integrating.

On performing the integration for each of the cases in (2)–(4), we then obtain antiderivatives $I_s^{(p,q)}$ that can be evaluated at the endpoints of each line segment in order to give a numerically computed line integral. In Cartesian geometry computing exact line integrals of polynomials is straightforward (see Bockman 1989); however, on the sphere the integration is nontrivial and is hence performed using the computational mathematics software Maple and then simplified by hand. In the following sections we provide $I_s^{(p,q)}$ for $p + q < 3$, as required for constructing remapping schemes up to third-order accuracy as well as the associated subgrid-cell reconstruction functions. Extensions to fourth order and beyond can be obtained via the process of integration also described below.

a. First-order scheme

We now turn our attention to the piecewise-constant GECoRe scheme as an example of the requisite calculations. Under this scheme, the subgrid-cell reconstruction for each cell on the source grid is given by the constant value $a_n^{(0,0)}$, that is,

$$\overline{f}_n = a_n^{(0,0)}, \tag{A1}$$

and hence the remapping scheme in (8) and (9) reduces to

$$\overline{f}_k = \frac{1}{A_k}\sum_{n=1}^{N} a_n^{(0,0)} \int_{A_{n,k}} dA. \tag{A2}$$

One can quickly observe that for this scheme we only require knowledge of the area covered by the overlapping regions, which is given by the interior integral term. Hence, in order to apply the divergence theorem and rewrite the area integral in terms of line integrals we require knowledge of the potential $\mathbf{\Psi}^{(0,0)}$ associated with the constant field $\phi = 1$. A simple calculation using (11) gives

$$\frac{\mathbf{\Psi}_\beta}{\rho\cos\alpha} = \frac{\tan\beta}{\rho}. \tag{A3}$$

The area integral is then evaluated via (17), where the antiderivatives $I_s^{(0,0)}$ follow from (18), and hence are given by

$$I_s^{(0,0)} = \int \frac{\tan\beta}{\rho}\, d\alpha. \tag{A4}$$

Integration of this quantity is then performed over each of the line segments described previously. For this constant field, we obtain the following closed-form relations.

Lines of constant β:

$$I_s^{(0,0)} = -\arccos(\sin\alpha \sin\beta). \tag{A5}$$

Lines of constant λ (panels 5 and 6):

$$I_s^{(0,0)} = \text{sign}(\sin\lambda \sin\theta)\arcsin(\cos\lambda \cos\alpha). \tag{A6}$$

Lines of constant θ (panels 1–4):

$$I_s^{(0,0)} = \alpha \sin\theta. \tag{A7}$$

Lines of constant θ (panels 5 and 6):

$$I_s^{(0,0)} = \text{sign}(\sin\theta)\arctan\left(\frac{\tan\alpha}{\tan\beta \sin\theta}\right) - \lambda \sin\theta. \tag{A8}$$

b. Second-order scheme

We now turn our attention to the higher-order GECoRe schemes, beginning with the second-order scheme, which is based on a piecewise-linear reconstruction. First, the antiderivatives associated with the background fields used in these high-order schemes are presented without derivation. Second, we provide a derivation of the reconstruction coefficients for this scheme.

1) ANTIDERIVATIVES

For the second-order-accurate reconstruction, following (16), we must find potentials $\mathbf{\Psi}^{(1,0)}$ and $\mathbf{\Psi}^{(0,1)}$ that satisfy

$$\nabla \cdot \boldsymbol{\Psi}^{(1,0)} = \tan\alpha \quad \text{and} \quad \nabla \cdot \boldsymbol{\Psi}^{(0,1)} = \tan\beta, \tag{A9}$$

where the divergence operator is defined in accordance with (11). From (16) we obtain

$$\frac{\Psi_\beta^{(1,0)}}{\rho\cos\alpha} = \frac{\tan\alpha\tan\beta}{\rho}, \quad \text{and} \quad \frac{\Psi_\beta^{(0,1)}}{\rho\cos\alpha} = -\frac{1}{\rho\cos^2\alpha}. \tag{A10}$$

The antiderivatives of these expressions then lead to $I_s^{(1,0)}$ and $I_s^{(0,1)}$, when evaluated along lines of constant β, constant latitude, and constant longitude (recall the antiderivatives along lines of constant α are zero).

Lines of constant β:

$$I_s^{(1,0)} = -\text{arcsinh}(\tan\beta\cos\alpha), \tag{A11}$$

$$I_s^{(0,1)} = -\text{arcsinh}(\tan\alpha\cos\beta). \tag{A12}$$

Lines of constant λ (panels 5 and 6):

$$I_s^{(1,0)} = \text{sign}(\sin\theta)[\text{arctanh}(\cos\lambda\cos\theta) - \cos\lambda\,\text{arctanh}(\cos\theta)], \tag{A13}$$

$$I_s^{(0,1)} = -\sin\lambda\ \text{arctanh}(\cos\theta). \tag{A14}$$

Lines of constant θ (panels 1–4):

$$I_s^{(1,0)} = -\sin\theta\ln(\cos\alpha), \tag{A15}$$

$$I_s^{(0,1)} = -\cos\theta\ln(\sec\alpha + \tan\alpha). \tag{A16}$$

Lines of constant θ (panels 5 and 6):

$$I_s^{(1,0)} = \text{sign}(\sin\theta)[-\cos\lambda\cos\theta + \text{arctanh}(\cos\lambda\cos\theta)], \tag{A17}$$

$$I_s^{(0,1)} = -\sin\lambda\cos\theta. \tag{A18}$$

2) Piecewise linear (second order) reconstruction

We can extend upon the first-order reconstruction by including a linear term in the subgrid-scale reconstruction. That is, for each cell n on the source grid, the field f_n, from (5), takes the following form:

$$f_n(x, y) = a_n^{(0,0)} + a_n^{(1,0)}(x - x_0) + a_n^{(0,1)}(y - y_0). \tag{A19}$$

For the second-order method, the reconstruction coefficients $a_n^{(i,j)}$ obtained from the Taylor series expansion are clearly the most natural choice. Recall that we can write the Taylor series expansion about $\mathbf{x}_0$ as[A1]

$$f_n = \overline{f}_n + \left(\frac{\partial f}{\partial x}\right)_n (x - x_0) + \left(\frac{\partial f}{\partial y}\right)_n (y - y_0). \tag{A20}$$

Hence, on comparing (A19) and (A20) we are inclined to choose the reconstruction coefficients according to

$$a_n^{(0,0)} = \overline{f}_n, \quad a_n^{(1,0)} = \left(\frac{\partial f}{\partial x}\right)_n, \quad a_n^{(0,1)} = \left(\frac{\partial f}{\partial y}\right)_n. \tag{A21}$$

It can quickly be verified that this choice satisfies the area-averaged field constraint in (7), and hence is a valid choice of reconstruction coefficients. Computationally, each of these coefficients can then be easily approximated via a discretized derivative operator. Numerous possibilities exist for the choice of discretized derivative operator that vary in both order and stencil size. We refer the reader to Chung (2002) for a list of possible discrete operators. The discretized derivative operator must be at least first-order accurate so as to obtain a second-order method, and must correspondingly increase in order for increasingly higher-order schemes.

Note that the second-order scheme requires knowledge of the area integrals (or line segment weights) for the fields $x = \tan\alpha$ and $y = \tan\beta$, in addition to the constant field, for a total of three weights per line segment.

c. *Third-order scheme*

In this section we provide the antiderivatives and reconstruction coefficients required for the third-order-accurate piecewise-parabolic scheme.

1) Antiderivatives

For the third-order-accurate reconstruction, following (16), we must find potentials $\boldsymbol{\Psi}^{(2,0)}$, $\boldsymbol{\Psi}^{(0,2)}$, and $\boldsymbol{\Psi}^{(1,1)}$ that satisfy

$$\nabla \cdot \boldsymbol{\Psi}^{(2,0)} = \tan^2\alpha, \quad \nabla \cdot \boldsymbol{\Psi}^{(0,2)} = \tan^2\beta, \quad \nabla \cdot \boldsymbol{\Psi}^{(1,1)} - \tan\alpha\tan\beta. \tag{A22}$$

From (16) we find

$$\frac{\Psi_\beta^{(2,0)}}{\rho\cos\alpha} = \frac{\tan^2\alpha\tan\beta}{\rho}, \tag{A23}$$

$$\frac{\Psi_\beta^{(0,2)}}{\rho\cos\alpha} = \frac{1}{\cos^2\alpha}\left[-\frac{\tan\beta}{\rho} + \text{arcsinh}(\tan\beta\cos\alpha)\right], \tag{A24}$$

[A1] Observe that (A20) is analogous to Eq. (5) in Jones (1999).

$$\frac{\Psi_\beta^{(1,1)}}{\rho\cos\alpha} = -\frac{\tan\alpha}{\rho\cos^2\alpha}. \tag{A25}$$

The antiderivatives of these expressions then lead to $I_s^{(2,0)}$, $I_s^{(0,2)}$, and $I_s^{(1,1)}$ when evaluated along the appropriate line segments.

Lines of constant β:

$$I_s^{(2,0)} = \tan\beta\ \mathrm{arcsinh}(\cos\beta\tan\alpha) + \arccos(\sin\alpha\sin\beta), \tag{A26}$$

$$I_s^{(0,2)} = \tan\alpha\ \mathrm{arcsinh}(\cos\alpha\tan\beta) + \arccos(\sin\alpha\sin\beta), \tag{A27}$$

$$I_s^{(1,1)} = -\rho. \tag{A28}$$

Lines of constant λ (panels 5 and 6):

$$I_s^{(2,0)} = -\frac{\sin\lambda\cos\lambda}{\sin\theta} + \arctan\left(\frac{\tan\lambda}{\sin\theta}\right), \tag{A29}$$

$$I_s^{(0,2)} = -\frac{\sin\lambda\cos\lambda}{\sin\theta} - \sin\lambda\cot\theta\ \mathrm{arctanh}(\cos\lambda\cos\theta) - \arctan(\cot\lambda\sin\theta), \tag{A30}$$

$$I_s^{(1,1)} = -\frac{\sin^2\lambda}{|\sin\theta|}. \tag{A31}$$

Lines of constant θ (panels 1–4):

$$I_s^{(2,0)} = \sin\theta(\tan\alpha - \alpha), \tag{A32}$$

$$I_s^{(0,2)} = \tan\alpha[-\sin\theta + \mathrm{arcsinh}(\tan\theta)], \tag{A33}$$

$$I_s^{(1,1)} = -\frac{\cos\theta}{\cos\alpha}. \tag{A34}$$

Lines of constant θ (panels 5 and 6):

$$I_s^{(2,0)} = -\frac{\cos^2\theta}{2\sin\theta}\sin\lambda\cos\lambda - \lambda\sin\theta\left(\frac{1}{2}\cos^2\theta + 1\right) + \arctan\left(\frac{\tan\lambda}{\sin\theta}\right), \tag{A35}$$

$$I_s^{(0,2)} = \frac{\cos^2\theta}{2\sin\theta}\sin\lambda\cos\lambda + \frac{\lambda}{\sin\theta}\left(\frac{1}{2}\cos^2\theta - 1\right) - \cot\theta\sin\lambda\ \mathrm{arctanh}(\cos\lambda\cos\theta) + \arctan\left(\frac{\tan\lambda}{\sin\theta}\right), \tag{A36}$$

$$I_s^{(1,1)} = -\frac{1}{2}|\sin\theta|\tan^2\alpha. \tag{A37}$$

2) PIECEWISE PARABOLIC (THIRD ORDER) RECONSTRUCTION

We can devise a third-order scheme by including parabolic terms in the subgrid-scale reconstruction. That is, for each cell n on the source grid, the field f_n takes the following form:

$$f_n(x, y) = a_n^{(0,0)} + a_n^{(1,0)}(x - x_0) + a_n^{(0,1)}(y - y_0) + a_n^{(2,0)}(x - x_0)^2 + a_n^{(1,1)}(x - x_0)(y - y_0) + a_n^{(0,2)}(y - y_0)^2. \tag{A38}$$

As with the second-order method, we begin by writing the Taylor series expansion of f_n about the centroid $\mathbf{x}_0$, obtaining

$$f_n(x, y) = f_n(\mathbf{x}_0) + \left(\frac{\partial f}{\partial x}\right)_n(x - x_0) + \left(\frac{\partial f}{\partial y}\right)_n(y - y_0) + \frac{1}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_n(x - x_0)^2 + \left(\frac{\partial^2 f}{\partial x\partial y}\right)_n(x - x_0)(y - y_0) + \frac{1}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_n(y - y_0)^2. \tag{A39}$$

Observe that in this expansion we have not fixed the value of the source field at $\mathbf{x}_0$ to be equal to $\overline{f}_n$. In fact, one can quickly verify that the choice $f_n(\mathbf{x}_0) = \overline{f}_n$ does not lead to a method consistent with the constraint in (7). The ''correct'' choice for $f_n(\mathbf{x}_0)$ is instead obtained by integrating (A39) over the source volume and rewriting the left-hand side in terms of the area-averaged field $\overline{f}_n$. Following this approach, we find[A2]

$$a_n^{(0,0)} = f_n(\mathbf{x}_0) = \overline{f}_n + \frac{1}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_n[x_0^2 - m_n^{(2,0)}] + \left(\frac{\partial^2 f}{\partial x\partial y}\right)_n[x_0 y_0 - m_n^{(1,1)}] + \frac{1}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_n[y_0^2 - m_n^{(0,2)}], \tag{A40}$$

where $m_n^{(p,q)}$ are the area-averaged moments, defined via

$$m_n^{(p,q)} = \frac{1}{A_n}\int_{A_n} x^p y^q\, dA. \tag{A41}$$

The remaining reconstruction coefficients are obtained from the Taylor expansion in (A39):

[A2] This choice reduces our reconstruction to the well-known PPM devised by Colella and Woodward (1984).

$$a_n^{(1,0)} = \left(\frac{\partial f}{\partial x}\right)_n, \quad a_n^{(0,1)} = \left(\frac{\partial f}{\partial y}\right)_n, \tag{A42}$$

$$a_n^{(2,0)} = \frac{1}{2}\left(\frac{\partial^2 f}{\partial x^2}\right)_n, \quad a_n^{(1,1)} = \left(\frac{\partial^2 f}{\partial x \partial y}\right)_n, \quad a_n^{(0,2)} = \frac{1}{2}\left(\frac{\partial^2 f}{\partial y^2}\right)_n. \tag{A43}$$

Each of these coefficients must be constructed from a discretized derivative operator of at least second order, and must increase in order correspondingly for higher-order schemes.

The third-order scheme requires knowledge of the area integrals, or line potentials, of the fields $x^2 = \tan^2\alpha$, $xy = \tan\alpha \tan\beta$, and $y^2 = \tan^2\beta$, in addition to all potentials from the first- and second-order schemes.

APPENDIX B

High-Order Bisected Element Reconstruction

Observe that for each of the high-order schemes described in appendix A the reconstruction is performed over the gnomonic coordinates, which are inherently discontinuous between panels. Hence, special consideration must be taken when performing remapping over an RLL source volume that covers two or more panels. The simplest solution to this problem is to divide the RLL cell into two or more cells, bisected by the panel edge. All high-order reconstruction coefficients can be maintained in this case, but the first-order cell average must be reevaluated in each subcell.

a. A second-order-accurate bisected element reconstruction

One can obtain a simple second-order approximation that conserves the scalar field by imposing

$$A_1\overline{f}_1 + A_2\overline{f}_2 = A_n\overline{f}_n, \tag{B1}$$

where $\overline{f}_1$ and $\overline{f}_2$ are the new area-averaged field values in the subcells spanning the panel edge and $\overline{f}_n$ is the area-averaged field in the original source cell. To solve this equation uniquely for $\overline{f}_1$ and $\overline{f}_2$, we also impose that the reconstructions in each subcell must be identical in RLL coordinates:.

$$\overline{f}_1 - \overline{f}_2 = \left(\frac{\partial f}{\partial \lambda}\right)_n(\lambda_1 - \lambda_2) + \left(\frac{\partial f}{\partial \theta}\right)_n(\theta_1 - \theta_2). \tag{B2}$$

These two conditions then lead to a simple second-order approximation that conserves the scalar field, given by

$$\overline{f}_1 = \overline{f}_n + \left(\frac{A_2}{A_n}\right)\left[\left(\frac{\partial f}{\partial \lambda}\right)_n(\lambda_1 - \lambda_2) + \left(\frac{\partial f}{\partial \theta}\right)_n(\theta_1 - \theta_2)\right] \tag{B3}$$

and

$$\overline{f}_2 = \overline{f}_n - \left(\frac{A_1}{A_n}\right)\left[\left(\frac{\partial f}{\partial \lambda}\right)_n(\lambda_1 - \lambda_2) + \left(\frac{\partial f}{\partial \theta}\right)_n(\theta_1 - \theta_2)\right]. \tag{B4}$$

Since the reconstruction in gnomonic coordinates requires that derivatives are aligned along x and y coordinate axes, additional work must be performed in order to rotate the reconstructed derivative in spherical coordinates to the gnomonic coordinate system. Appendix C gives equations for translating the reconstructed derivatives in gnomonic coordinates from a reconstruction in RLL coordinates for both the second- and third-order schemes.

b. A third-order-accurate bisected element reconstruction

A third-order-accurate reconstruction for a bisected cell can be obtained as an extension of the method discussed in the previous section, and is necessary for remapping from the RLL grid to CS grid using the PPM scheme. Henceforth, we will use the notation (x_i, y_i) to denote the centroid of each subcell ($i \in \{1, 2\}$) in gnomonic coordinates and (λ_i, θ_i) to denote the centroid in RLL coordinates.

In this case, the second derivatives of the field are inherited from the parent cell in each subcell, but the cell-averaged values and first derivatives must be computed separately. Using a simple Taylor series expansion in RLL coordinates, we can write the subcell first derivatives as

$$\left(\frac{\partial f}{\partial \lambda}\right)_i = \left(\frac{\partial f}{\partial \lambda}\right)_n + \left(\frac{\partial^2 f}{\partial \lambda \partial \theta}\right)_n(\theta_i - \theta_n) + \left(\frac{\partial^2 f}{\partial \lambda^2}\right)_n(\lambda_i - \lambda_n) \tag{B5}$$

and

$$\left(\frac{\partial f}{\partial \theta}\right)_i = \left(\frac{\partial f}{\partial \theta}\right)_n + \left(\frac{\partial^2 f}{\partial \lambda \partial \theta}\right)_n(\lambda_i - \lambda_n) + \left(\frac{\partial^2 f}{\partial \theta^2}\right)_n(\theta_i - \theta_n). \tag{B6}$$

However, one cannot simply use a Taylor series to obtain $f(\mathbf{x}_0)$, since we have no guarantee that it would satisfy the conservation constraint given in (B1). Instead, we impose (B1), which, in conjunction with the area-averaged constraint in (A40), gives

$$A_1 a_1^{(0,0)} + A_2 a_2^{(0,0)} = \overline{f}_n (A_1 + A_2) + A_1 \left[\frac{1}{2} \left(\frac{\partial^2 f}{\partial x^2} \right)_1 \{ [x_1^2 - m_1^{(2,0)}] \} + \left(\frac{\partial^2 f}{\partial x \partial y} \right)_1 [x_1 y_1 - m_1^{(1,1)}] + \frac{1}{2} \left(\frac{\partial^2 f}{\partial y^2} \right)_1 [y_1^2 - m_1^{(0,2)}] \right] + A_2 \left[\frac{1}{2} \left(\frac{\partial^2 f}{\partial x^2} \right)_2 [x_2^2 - m_2^{(2,0)}] + \left(\frac{\partial^2 f}{\partial x \partial y} \right)_2 [x_2 y_2 - m_2^{(1,1)}] + \frac{1}{2} \left(\frac{\partial^2 f}{\partial y^2} \right)_2 [y_2^2 - m_2^{(0,2)}] \right]. \quad \text{(B7)}$$

A second equation can be obtained by imposing that the subcells have the same reconstruction in RLL coordinates, which can be expressed as

$$a_2^{(0,0)} - a_1^{(0,0)} = \left[\left(\frac{\partial f}{\partial \lambda} \right)_2 \lambda_2 - \left(\frac{\partial f}{\partial \lambda} \right)_1 \lambda_1 \right] + \left[\left(\frac{\partial f}{\partial \theta} \right)_2 \theta_2 - \left(\frac{\partial f}{\partial \theta} \right)_1 \theta_1 \right] - \frac{1}{2} \left(\frac{\partial^2 f}{\partial \lambda^2} \right) (\lambda_2^2 - \lambda_1^2) - \left(\frac{\partial^2 f}{\partial \lambda \partial \theta} \right) (\lambda_2 \theta_2 - \lambda_1 \theta_1) - \frac{1}{2} \left(\frac{\partial^2 f}{\partial \theta^2} \right) (\theta_2^2 - \theta_1^2). \quad \text{(B8)}$$

If we define the right-hand side of (B7) as c_A and the right-hand side of (B8) as c_B, then the $a_i^{(0,0)}$ in each subcell take the following form:

$$a_1^{(0,0)} = \frac{c_A - A_2 c_B}{A_1 + A_2}, \quad \text{and} \quad a_2^{(0,0)} = \frac{c_A + A_1 c_B}{A_1 + A_2}. \quad \text{(B9)}$$

The reconstruction coefficients $a_i^{(0,0)}$ can then be converted back to area-averaged values $\overline{f}_i$ on applying (7).

APPENDIX C

The Gnomonic Cubed-Sphere Projection

A point on the cubed sphere in the gnomonic projection is normally given in terms of (x, y, n_p) coordinates, where $x, y \in [-1, 1]$ and $n_p \in \{1, 2, 3, 4, 5, 6\}$. By convention, we choose panels 1–4 to be along the spherical equator, with panels 5 and 6 centered on the southern and northern pole, respectively. As in (1), gnomonic coordinates are related to equiangular coordinates via the relations

$$x = a \tan\alpha \quad \text{and} \quad y = a \tan\beta, \quad \text{(C1)}$$

where, without loss of generality, we have chosen $a = 1$. Since gnomonic coordinates are panel dependent, the change of coordinates relations are dependent on the choice of panel.

a. Panels 1–4 (equatorial panels)

In terms of spherical coordinates, x and y take the following form:

$$x = \tan\lambda^* \quad \text{and} \quad y = \tan\theta \sec\lambda^*, \quad \text{(C2)}$$

where λ^* is the panel-centric longitude coordinate, defined in terms of the panel k by

$$\lambda^* = \lambda - \frac{\pi}{2}(k - 1). \quad \text{(C3)}$$

Inverting (C2) yields

$$\lambda^* = \arctan x \quad \text{and} \quad \theta = \arctan\left(\frac{y}{\sqrt{1 + x^2}} \right). \quad \text{(C4)}$$

b. Panels 5 and 6 (polar panels)

For simplicity, we define a panel indicator variable as

$$k = \text{sign}(\theta). \quad \text{(C5)}$$

Observe that on the south polar panel in (5) and the north polar panel in (6), the indicator variables takes on the values -1 and $+1$, respectively, over the entire panel.

In terms of spherical coordinates, x and y take the following form:

$$x = k \sin\lambda \cot\theta \quad \text{and} \quad y = -\cos\lambda \cot\theta. \quad \text{(C6)}$$

Inverting (C6) yields

$$\lambda = -k \arctan(x/y) \quad \text{and} \quad \theta = k \arctan\left(\frac{1}{\sqrt{x^2 + y^2}} \right). \quad \text{(C7)}$$

c. Change-of-coordinates matrices for high-order schemes

When performing interpolation between RLL and cubed-sphere grids, we are required to obtain values for the reconstruction coefficients in terms of gnomonic coordinates. Since obtaining the reconstructed derivatives in RLL coordinates is relatively simple, a quick application of the chain rule leads to a change-of-coordinates matrix of the following form:

$$\begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \lambda}{\partial x} & \frac{\partial \theta}{\partial x} \\ \frac{\partial \lambda}{\partial y} & \frac{\partial \theta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial f}{\partial \lambda} \\ \frac{\partial f}{\partial \theta} \end{bmatrix}. \quad \text{(C8)}$$

Hence, the following matrices allow us to rotate the reconstructed derivatives in RLL coordinates, which can be easily obtained via a discretized derivative operator, to gnomonic coordinates.

Panels 1–4 (equatorial panels):

$$\begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} \cos^2\lambda & -\dfrac{1}{4}\sin(2\lambda)\sin(2\theta) \\ 0 & \cos\lambda\cos^2\theta \end{bmatrix} \begin{bmatrix} \dfrac{\partial f}{\partial \lambda} \\ \dfrac{\partial f}{\partial \theta} \end{bmatrix}, \quad \text{(C9)}$$

Panels 5 and 6 (polar panels):

$$\begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} k\cos\lambda\tan\theta & -k\sin\lambda\sin^2\theta \\ \sin\lambda\tan\theta & \cos\lambda\sin^2\theta \end{bmatrix} \begin{bmatrix} \dfrac{\partial f}{\partial \lambda} \\ \dfrac{\partial f}{\partial \theta} \end{bmatrix}. \quad \text{(C10)}$$

For third and higher order schemes, we also require equations that express the second-order gnomonic derivatives

$$\frac{\partial^2 f}{\partial x^2}, \quad \frac{\partial^2 f}{\partial x \partial y}, \quad \text{and} \quad \frac{\partial^2 f}{\partial y^2}$$

in terms of derivatives in RLL coordinates. Again, applying the chain rule, we find

$$\begin{bmatrix} \dfrac{\partial^2 f}{\partial x^2} \\ \dfrac{\partial^2 f}{\partial x \partial y} \\ \dfrac{\partial^2 f}{\partial y^2} \end{bmatrix} = \underbrace{\begin{bmatrix} \dfrac{\partial^2 \lambda}{\partial x^2} & \dfrac{\partial^2 \theta}{\partial x^2} \\ \dfrac{\partial^2 \lambda}{\partial x \partial y} & \dfrac{\partial^2 \theta}{\partial x \partial y} \\ \dfrac{\partial^2 \lambda}{\partial y^2} & \dfrac{\partial^2 \theta}{\partial y^2} \end{bmatrix}}_{\mathbf{A}^{(1,2)}} \begin{bmatrix} \dfrac{\partial f}{\partial \lambda} \\ \dfrac{\partial f}{\partial \theta} \end{bmatrix} + \underbrace{\begin{bmatrix} \left(\dfrac{\partial \lambda}{\partial x}\right)^2 & 2\dfrac{\partial \lambda}{\partial x}\dfrac{\partial \theta}{\partial x} & \left(\dfrac{\partial \theta}{\partial x}\right)^2 \\ \dfrac{\partial \lambda}{\partial x}\dfrac{\partial \lambda}{\partial y} & \dfrac{\partial \theta}{\partial x}\dfrac{\partial \lambda}{\partial y} + \dfrac{\partial \theta}{\partial x}\dfrac{\partial \lambda}{\partial y} & \dfrac{\partial \theta}{\partial y}\dfrac{\partial \theta}{\partial x} \\ \left(\dfrac{\partial \lambda}{\partial y}\right)^2 & 2\dfrac{\partial \lambda}{\partial y}\dfrac{\partial \theta}{\partial y} & \left(\dfrac{\partial \theta}{\partial y}\right)^2 \end{bmatrix}}_{\mathbf{A}^{(2,2)}} \begin{bmatrix} \dfrac{\partial^2 f}{\partial \lambda^2} \\ \dfrac{\partial^2 f}{\partial \lambda \partial \theta} \\ \dfrac{\partial^2 f}{\partial \theta^2} \end{bmatrix}. \quad \text{(C11)}$$

Evaluating the matrix $\mathbf{A}^{(1,2)}$ on panels 1–4 gives

$$\mathbf{A}^{(1,2)} = \begin{bmatrix} -\cos^2\lambda\sin(2\lambda) & -\dfrac{1}{2}\cos^2\lambda\sin(2\theta)[\cos(2\lambda) - \sin^2\lambda\cos(2\theta)] \\ 0 & -\dfrac{1}{2}\cos\lambda\sin(2\lambda)\cos^2\theta\cos(2\theta) \\ 0 & -\cos^2\lambda\sin(2\lambda)\cos^2\theta \end{bmatrix}. \quad \text{(C12)}$$

On panels 5 and 6 we obtain

$$\mathbf{A}^{(1,2)} = \begin{bmatrix} -\sin(2\lambda)\tan^2\theta & -\sin^2\theta[\cos^2\lambda\tan\theta - \sin^2\lambda\sin(2\theta)] \\ k\cos(2\lambda)\tan^2\theta & -\dfrac{1}{2}k\sin(2\lambda)\sin^2\theta[\tan\theta + \sin(2\theta)] \\ \sin(2\lambda)\tan^2\theta & -\sin^2\theta[\sin^2\lambda\tan\theta - \cos^2\lambda\sin(2\theta)] \end{bmatrix}. \quad \text{(C13)}$$

The matrix $\mathbf{A}^{(2,2)}$ is trivial to calculate, given (C9) and (C10), and hence is not provided here.

APPENDIX D

The Equiangular Cubed-Sphere Projection

In this appendix we briefly provide details on coordinate relations and the metric associated with the equiangular cubed-sphere projection. For a more thorough treatment of this material, we refer to Nair et al. (2005).

A point on the cubed sphere in the equiangular projection is normally given in terms of (α, β, n_p) (ABP) coordinates, where $\alpha, \beta \in [-\pi/4, \pi/4]$ and $n_p \in \{1, 2, 3, 4, 5, 6\}$. By convention, we choose panels 1–4 to be along the spherical equator, with panels 5 and 6 centered on the southern and northern pole, respectively.

a. Panels 1–4 (equatorial panels)

In terms of spherical coordinates, α and β take the following form:

$$\alpha = \lambda^*, \quad \beta = \arctan(\tan\theta\sec\lambda^*), \quad \text{(D1)}$$

where λ^* is the panel-centric longitude coordinate, defined earlier in (C3). Inverting (D1) yields

$$\lambda^* = \alpha, \quad \theta = \arctan(\tan\beta\cos\alpha). \quad \text{(D2)}$$

b. Panels 5 and 6 (polar panels)

In terms of spherical coordinates, α and β take the following form:

$$\alpha = k\arctan(\sin\lambda\cot\theta), \quad \beta = -\arctan(\cos\lambda\cot\theta). \tag{D3}$$

Inverting (D3) yields

$$\lambda = -k\arctan\left(\frac{\tan\alpha}{\tan\beta}\right), \quad \theta = -\arctan\left(\frac{\cos\lambda}{\tan\beta}\right). \tag{D4}$$

A useful identity that follows from these relationships is

$$\tan^2\alpha + \tan^2\beta = \cot^2\theta. \tag{D5}$$

c. The equiangular cubed-sphere metric

The equiangular cubed-sphere metric is given by

$$g_{i,j} = \frac{1}{\rho^4\cos^2\alpha\cos^2\beta}\begin{bmatrix} 1+\tan^2\alpha & -\tan\alpha\tan\beta \\ -\tan\alpha\tan\beta & 1+\tan^2\beta \end{bmatrix}, \tag{D6}$$

where ρ is defined by

$$\rho^2 = 1 + \tan^2\alpha + \tan^2\beta. \tag{D7}$$

The volume element for this metric is then

$$\sqrt{g} = \sqrt{\det(g_{i,j})} = (\rho^3\cos^2\alpha\cos^2\beta)^{-1}. \tag{D8}$$

We can use the metric to define unit basis vectors, such that $\mathbf{e}\cdot\mathbf{e} = g_{i,j}e^ie^j = 1$. In terms of the natural basis $\hat{\alpha}$ and $\hat{\beta}$, these can be written as

$$(\mathbf{e}_\alpha) = (\rho^2\cos^2\alpha\cos\beta)\hat{\alpha}, \quad (\mathbf{e}_\beta) = (\rho^2\cos\alpha\cos^2\beta)\hat{\beta}. \tag{D9}$$

The nonorthogonality parameter, which determines the degree or nonorthogonality of the basis vectors at each point on the manifold, is then given by

$$\cos(\phi) = \mathbf{e}_\alpha\cdot\mathbf{e}_\beta = -\sin\alpha\sin\beta. \tag{D10}$$

REFERENCES

Barth, T., and D. Jespersen, 1989: The design and application of upwind schemes on unstructured meshes. *Proc. AIAA 27th Aerospace Sciences Meeting,* Reno, NV, AIAA, AIAA–89–0366.

Bockman, S., 1989: Generalizing the formula for areas of polygons to moments. *Amer. Math. Mon.,* **96** (2), 131–132.

Chung, T. J., 2002: *Computational Fluid Dynamics*. Cambridge University Press, 800 pp.

Colella, P., and P. R. Woodward, 1984: The piecewise parabolic method (PPM) for gas-dynamical simulations. *J. Comput. Phys.,* **54,** 174–201.

Collins, W., and Coauthors, 2006: The Community Climate System Model version 3 (CCSM3). *J. Climate,* **19,** 2122–2143.

Dukowicz, J. K., and J. W. Kodis, 1987: Accurate conservative remapping (rezoning) for arbitrary Lagrangian-Eulerian computations. *SIAM J. Sci. Stat. Comput.,* **8** (3), 305–321.

Jones, P. W., 1999: First- and second-order conservative remapping schemes for grids in spherical coordinates. *Mon. Wea. Rev.,* **127,** 2204–2210.

Lauritzen, P. H., and R. D. Nair, 2008: Monotone and conservative cascade remapping between spherical grids (CaRS): Regular latitude–longitude and cubed-sphere grids. *Mon. Wea. Rev.,* **136,** 1416–1432.

Margolin, L. G., and M. Shashkov, 2003: Second-order sign-preserving conservative interpolation (remapping) on general grids. *J. Comput. Phys.,* **184,** 266–298.

Nair, R. D., and B. Machenhauer, 2002: The mass-conservative cell-integrated semi-Lagrangian advection scheme on the sphere. *Mon. Wea. Rev.,* **130,** 649–667.

——, S. J. Thomas, and R. D. Loft, 2005: A discontinuous Galerkin transport scheme on the cubed sphere. *Mon. Wea. Rev.,* **133,** 814–828.

Putman, W. M., and S.-J. Lin, 2007: Finite-volume transport on various cubed-sphere grids. *J. Comput. Phys.,* **227,** 55–78.

Rančić, M., 1992: Semi-Lagrangian piecewise biparabolic scheme for two-dimensional horizontal advection of a passive scalar. *Mon. Wea. Rev.,* **120,** 1394–1405.

——, J. Purser, and F. Mesinger, 1996: A global shallow water model using an expanded spherical cube. *Quart. J. Roy. Meteor. Soc.,* **122,** 959–982.

Ronchi, C., R. Iacono, and P. S. 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.

Zerroukat, M., N. Wood, and A. Staniforth, 2005: A monotonic and positive-definite filter for a semi-Lagrangian inherently conserving and efficient (SLICE) scheme. *Quart. J. Roy. Meteor. Soc.,* **131** (611), 2923–2936.