A Discontinuous Galerkin Transport Scheme on the Cubed Sphere

RAMACHANDRAN D. NAIR, STEPHEN J. THOMAS, AND RICHARD D. LOFT

Scientific Computing Division, National Center for Atmospheric Research,* Boulder, Colorado

ABSTRACT

A conservative transport scheme based on the discontinuous Galerkin (DG) method has been developed for the cubed sphere. Two different central projection methods, equidistant and equiangular, are employed for mapping between the inscribed cube and the sphere. These mappings divide the spherical surface into six identical subdomains, and the resulting grid is free from singularities. Two standard advection tests, solid-body rotation and deformational flow, were performed to evaluate the DG scheme. Time integration relies on a third-order total variation diminishing (TVD) Runge–Kutta scheme without a limiter. The numerical solutions are accurate and neither exhibit shocks nor discontinuities at cube-face edges and vertices. The numerical results are either comparable or better than a standard spectral element method. In particular, it was found that the standard relative error metrics are significantly smaller for the equiangular as opposed to the equidistant projection.

1. Introduction

Recent paradigm shifts in large-scale scientific computing have motivated investigations into numerical methods that are more suitable for distributed-memory parallel computers. In atmospheric modeling, global spectral methods have dominated for the past two decades in weather and climate simulation. However, global methods based upon the spherical harmonic basis functions require expensive nonlocal communication operations and thus have difficulty in exploiting the full potential of current high-performance parallel computers. In recent years, research has focused on local methods such as spectral elements (e.g., Taylor et al. 1997; Thomas and Loft 2002; Giraldo and Rosmond 2004; Fournier et al. 2004) and alternatives to the standard latitude–longitude grid in spherical geometry. Of particular interest are geometries that are free from singularites at the poles, such as the cubed sphere (Ronchi et al. 1996; Rančić et al. 1996) and geodesic grids (Randall et al. 2002). Geometric flexibilty in combination with high-order numerics is also desirable in order to support local grid refinement and adaptive solution techniques. Indeed, these are the main advantages of

the discontinuous Galerkin (DG) method. In addition, the DG scheme is both globally and locally conservative, making it an ideal candidate for atmospheric modeling applications. Thus, in this paper, a discontinuous Galerkin transport scheme is developed and applied to the 2D mass continuity equation, expressed in generalized curvilinear coordinates on the cubed sphere.

The discontinuous Galerkin method is a type of finite-element method that permits discontinuities of the numerical solution to exist at interelement interfaces. The Riemann problem at an element interface is solved using techniques found in the finite-volume literature. The DG method was originally introduced by Reed and Hill (1973) and later mathematically analyzed by Lesaint and Raviart (1974) for the linear advection equation. In a series of papers, Cockburn and Shu (1989, 1998) developed a high-order accurate total variation bounded Runge–Kutta discontinuous Galerkin (RKDG) method for the solution of nonlinear systems of conservation laws. They introduced Runge–Kutta (RK) integration for the DG method combined with slope limiters to control spurious oscillations. The resulting scheme was shown to be formally high-order accurate and nonlinearly stable. The DG method became very popular in different branches of computational science after the introduction of RKDG. A comprehensive review of RKDG is given in Cockburn and Shu (2001), and examples of applications can be found in Cockburn et al. (2000).

Recently, Giraldo et al. (2002) developed a global DG shallow water model using the constrained 3D equations of motion in Cartesian coordinates and dem-

---

Corresponding author address: Ramachandran D. Nair, Scientific Computing Division, National Center for Atmospheric Research, 1850 Table Mesa Drive, Boulder, CO 80305.
E-mail: rnair@ucar.edu

onstrated its application on an icosahedral grid. However, the DG method on the sphere considered here is significantly different from Giraldo et al. (2002). The main differences include the issues of basic formulation, spatial discretization, and time integration. Giraldo et al. (2002) have used the 3D Cartesian equations to solve a 2D problem on the sphere. This formulation not only increases the dimensionality of the problem but also imposes an additional constraint on the system of equations. We have employed a 2D formulation in curvilinear coordinates (tensor form). Our implementation of a DG scheme is on the cubed sphere, and equiangular (central) projection results in a much more uniform distribution of grid points. The DG scheme in Giraldo et al. (2002) employs an icosahedral grid and a *nodal* expansion for the spatial discretization. We instead use a *modal* expansion, namely because it is straightforward to incorporate element-based (adaptive) limiting and *p* refinement, in contrast with the nodal expansion approach (Henderson 1999). We use a third-order total variation diminishing (TVD) Runge–Kutta scheme for time integration, whereas Giraldo et al. (2002) apply a fourth-order non-TVD–RK method and optionally use a filter to reduce oscillations. No filters or additional dissipation mechanisms of any kind are applied during the time integration. We adopt a TVD–RK scheme because this will help us to develop a fully TVD approach for future work. A TVD scheme preserves strong stability (Gottlieb et al. 2001), implying that no new local minima or maxima will be generated.

This paper is organized as follows: In section 2 we describe the basic DG discretization in 1D, and the extension to two dimensions is presented in section 3. Section 4 describes the implementation of DG on the cubed sphere. Numerical experiments are presented in section 5, and these are followed by conclusions in section 6.

2. DG in one dimension

To introduce the basic DG discretization and associated notation, consider the following initial value problem for a one-dimensional scalar conservation law:

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} = 0 \text{ in } \Omega \times (0, T), \quad (1)$$

with the initial condition

$$U_0(x) = U(x, t = 0), \forall x \in \Omega.$$

It is assumed that the spatial domain Ω is periodic and partitioned into nonoverlapping elements (intervals) $I_j = [x_{j-1/2}, x_{j+1/2}], j = 1, \ldots, N_j$. Let $\Delta x_j = x_{j+1/2} - x_{j-1/2}$ be the width of an element and define the midpoint of an interval to be $x_j = (x_{j+1/2} + x_{j-1/2})/2$.

A weak formulation of the problem is obtained by multiplying (1) by an arbitrary *smooth* test function $\varphi(x)$ and integrating over an element I_j,

$$\int_{I_j} \left[\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} \right] \varphi(x) dx = 0. \quad (2)$$

Integrating the second term of (2) by parts yields

$$\int_{I_j} \frac{\partial U(x, t)}{\partial t} \varphi(x) dx - \int_{I_j} F[U(x, t)] \frac{\partial \varphi(x)}{\partial x} dx$$
$$+ F[U(x_{j+1/2}, t)] \varphi(x_{j+1/2}^-) - F[U(x_{j-1/2}, t)] \varphi(x_{j-1/2}^+) = 0, \quad (3)$$

where $\varphi(x_{j+1/2}^-)$ and $\varphi(x_{j-1/2}^+)$ are the values of the function $\varphi(x)$ at the end points $x_{j+1/2}$ and $x_{j-1/2}$ of the element I_j, respectively.

At an interface between elements (e.g., the points $x_{j\pm1/2}$), the flux function F is not uniquely defined, and a suitable numerical flux must be determined according to the classical finite-volume method. For example, the nonlinear flux function $F[U(x_{j+1/2}, t)]$ is replaced by a *numerical* flux $\hat{F}(U)_{j+1/2}(t)$ that depends on two values, the left and right limits of the discontinuous function U evaluated at the interface $x_{j+1/2}$ such that

$$\hat{F}(U)_{j+1/2}(t) = \hat{F}[U(x_{j+1/2}^-, t), U(x_{j+1/2}^+, t)]. \quad (4)$$

A variety of numerical fluxes are available to approximate the solution of the resulting Riemann problem (see Cockburn and Shu 2001). However, the Lax–Friedrichs numerical flux was chosen for the present study because of its simplicity:

$$\hat{F}(a, b) = \frac{1}{2}[F(a) + F(b) - \overline{\alpha}(b - a)], \quad (5)$$

where $\overline{\alpha}$ is specified as the upper bound on $|F'(U)|$ for a scalar problem.

a. Space discretization

Let V_h^k be a finite dimensional space such that $V_h^k = \{p:p|_{I_j} \in \mathbb{P}_k(I_j)\}$ where $\mathbb{P}_k(I_j)$ is the space of polynomials in I_j of degree at most k, $\forall j = 1, \ldots, N_j$. The algorithm for solving (3) involves finding an approximate solution $U_h(x, t) \approx U(x, t)$, for each time $t \in (0, T)$, such that the *test* function $\varphi_h = \varphi$ belongs to V_h^k. In a Galerkin finite-element method, the test functions and the solution both come from the same space.

For the approximate solution $U_h(x, t)$, the DG space discretization based on the weak formulation (3) is written as follows:

$$\int_{I_j} \frac{\partial U_h(x, t)}{\partial t} \varphi_h(x) dx = \int_{I_j} F[U_h(x, t)] \frac{\partial \varphi_h(x)}{\partial x} dx$$
$$- [\hat{F}(U_h)_{j+1/2}(t) \varphi_h(x_{j+1/2}^-)$$
$$+ \hat{F}(U_h)_{j-1/2}(t) \varphi_h(x_{j-1/2}^+)], \quad (6)$$

where $\varphi_h \in V_h^k, \forall I_j, j = 1, \ldots, N_j$.

A high-order Gaussian quadrature rule is applied to evaluate the integral on the right-hand side of (6) and the numerical flux term is computed according to (5). A local coordinate variable ξ is defined by an affine transformation mapping an element I_j onto $[-1,1]$,

$$\xi = \frac{2(x - x_j)}{\Delta x_j} \Rightarrow dx = \frac{\Delta x_j}{2} d\xi, \quad \frac{\partial}{\partial x} = \frac{2}{\Delta x_j} \frac{\partial}{\partial \xi}. \quad (7)$$

The natural choice for the set of basis functions spanning the space V_h^k are the Legendre polynomials, $\mathcal{B} = \{P_\ell(\xi), \ell = 0, 1, \ldots, k\}$. A major advantage of this choice is that the computations associated with (6) are greatly simplified by the L^2 orthogonality properties of Legendre polynomials. The approximate solution in element I_j is denoted $U_j(\xi, t)$, and is expressed in terms of the variable ξ,

$$U_j(\xi, t) = \sum_{\ell=0}^{k} U_j^\ell(t) P_\ell(\xi) \quad \text{for} \quad \xi \in [-1, 1]. \quad (8)$$

The orthogonality of $P_\ell(\xi)$ implies that

$$\int_{-1}^{1} P_\ell(\xi) P_m(\xi) \, d\xi = \frac{2}{2\ell + 1} \delta_{\ell m}, \quad (9)$$

and the expansion coefficients (or degrees of freedom) U_j^ℓ in (8) are given by

$$U_j^\ell(t) = \frac{2\ell + 1}{2} \int_{-1}^{1} U_j(\xi, t) P_\ell(\xi) \, d\xi \, \ell = 0, 1, \ldots, k. \quad (10)$$

To obtain a simplified form for (6), replace φ_h and U_h in (6) by $P_\ell(\xi)$ and apply the summation in (8). Next, by employing the transformation (7), the relations $P_\ell(1) = 1$, $P_\ell(-1) = (-1)^\ell$, and (9), it follows that

$$\frac{d}{dt} U_j^\ell(t) = \frac{2\ell + 1}{\Delta x_j} \left[\int_{-1}^{1} F[U_j(\xi, t)] \frac{\partial P_\ell(\xi)}{\partial \xi} \, d\xi \right.$$
$$\left. - \{\hat{F}[U_j(1)](t) - (-1)^\ell \hat{F}[U_j(-1)](t)\} \right]. \quad (11)$$

The initial values of $U_j^\ell(t = 0)$ are obtained by projecting the initial condition $U_0[x(\xi)]$ onto V_h^k using (10).

The basis considered above is often referred to as a *modal* basis. In this case, there is a hierarchy of modes—starting with the constant mode, proceeding with the linear, the quadratic, and the cubic up to the kth mode. The kth mode represents the polynomial of maximum degree k in $\mathcal{B}$. Such a basis can easily accomodate hierarchial p refinement by increasing the order of the polynomial. Moreover, element-wise adaptive limiting (Biswas et al. 1994) can be easily incorporated for the DG method based on a modal basis. An alternative choice is a *nodal* basis, which is widely used in spectral element (SE) methods. This basis set is con-

structed utilizing the Lagrangian interpolants of degree k on a Gauss–Lobatto–Legendre (GLL) grid. Giraldo et al. (2002) have used a nodal basis for their DG method. However, the choice of a nodal basis makes the DG method less flexible as far as adaptive limiting and p refinement are concerned. The relative merits of these two approaches are discussed in Karniadakis and Sherwin (1999) and Henderson (1999).

b. Time integration

The semidiscretized Eq. (11) is an ordinary differential equation (ODE) that can be written in the more compact form

$$\frac{d}{dt} U = L(U) \text{ in } (0, T). \quad (12)$$

A number of time integrators are available for solving (12). A TVD–RK scheme can remove oscillations due to time discretization. For the present study we adopt a third-order TVD–RK scheme (Shu 1988; Gottlieb et al. 2001) as follows:

$$U^{(1)} = U^n + \Delta t L(U^n),$$
$$U^{(2)} = \frac{3}{4} U^n + \frac{1}{4} U^{(1)} + \frac{1}{4} \Delta t L(U^{(1)}),$$
$$U^{n+1} = \frac{1}{3} U^n + \frac{2}{3} U^{(2)} + \frac{2}{3} \Delta t L(U^{(2)}), \quad (13)$$

where the superscripts n and $n + 1$ denote time levels t and $t + \Delta t$, respectively. The above scheme is suitable when the solution is smooth. However, if the solution contains strong shocks or discontinuities, oscillations that lead to nonlinear instabilities will appear. A TVD–RK time-integration scheme cannot alone control such undesirable effects, and a slope limiter is required after each step of the RK time integration (13). Minmod-type limiters (Cockburn and Shu 1989) and moment-based limiters (Biswas et al. 1994) are typically used to control these oscillations. Slope-limiting algorithms have been developed for finite-volume applications and these could be extended to DG methods. Limiters and filters have not been applied in the present study.

The Courant number for the DG scheme is estimated to be $1/(2k + 1)$, where k is the degree of the polynomial; however, no theoretical proof exists when $k > 1$ (see Cockburn and Shu 1989).

3. DG in two dimensions

Now consider the following 2D scalar conservation law:

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{F}(U) = 0, \text{ in } \Omega \times (0, T), \quad (14)$$

for all $(x, y) \in \Omega$. Here, $\nabla = (\partial/\partial x, \partial/\partial y)$ is the 2D gradient operator. The flux function is $\mathbf{F} = (F, G)$ and

$U = U(x, y, t)$. The initial condition for the problem is $U_0(x, y) = U(x, y, t = 0)$, and the rectangular domain Ω is periodic in both the x and y directions. Following the derivation of the previous section, extension of the 1D discretization to 2D is straightforward.

The domain Ω is partitioned into $N_x \times N_y$ rectangular nonoverlapping elements Ω_{ij} such that

$$\Omega_{ij} = \{(x, y)|x \in [x_{i-1/2}, x_{i+1/2}], y \in [y_{j-1/2}, y_{j+1/2}]\},$$

$$(15)$$

for $i = 1, 2, \ldots, N_x; j = 1, 2, \ldots, N_y$.

The size of an element Ω_{ij} is determined by $\Delta x_i = (x_{i+1/2} - x_{i-1/2})$ and $\Delta y_j = (y_{j+1/2} - y_{j-1/2})$ in the x and y directions, respectively. Consider an element Ω_{ij} in the partition of Ω and an approximate solution $U_h \in V_h^k$. Multiplication of (14) by a test function $\varphi_h(x, y) \in V_h^k$ and integration over the element Ω_{ij} results in a weak Galerkin formulation of the problem, analogous to (3):

$$\frac{\partial}{\partial t} \int_{\Omega_{ij}} U_h(x, y, t)\, \varphi_h(x, y)\, d\Omega$$

$$- \int_{\Omega_{ij}} \mathbf{F}[U_h(x, y, t)] \cdot \nabla \varphi(x, y)\, d\Omega$$

$$+ \int_{\partial\Omega_{ij}} \mathbf{F}[U_h(x, y, t)] \cdot \mathbf{n}\, \varphi_h(x, y) ds = 0, \quad (16)$$

where $\mathbf{n}$ is the outward-facing unit normal vector on the element boundary $\partial\Omega_{ij}$. The analytic flux $\mathbf{F}(U_h) \cdot \mathbf{n}$ in (16) is replaced by a specific numerical flux $\hat{\mathbf{F}}(U_h^-, U_h^+)$ such as the Lax–Friedrichs flux (5). The numerical flux resolves the discontinuity along the element edges and provides the only mechanism by which adjacent elements interact.

A set of local orthogonal basis function $\mathcal{B}$ and local variables (ξ, η) for each element are chosen as in the 1D case

$$\xi = \frac{2(x - x_i)}{\Delta x_i}, \quad \eta = \frac{2(y - y_j)}{\Delta y_j}, \quad (17)$$

where $x_i = (x_{i+1/2} + x_{i-1/2})/2$, $y_j = (y_{j+1/2} + y_{j-1/2})/2$. The transformation (17) maps Ω_{ij} onto the reference element $\tilde{\Omega}_{ij} \equiv [-1, 1] \otimes [-1, 1]$. In the (ξ, η) coordinate system, the test function is chosen to be a tensor product of the Legendre polynomials $P_\ell(\xi)\, P_m(\eta)$, and the approximate solution $U_{ij}(\xi, \eta, t)$ is expanded in terms of the basis functions in $\mathcal{B}$,

$$U_{ij}(\xi, \eta, t) = \sum_{\ell=0}^{k} \sum_{m=0}^{k} \hat{U}_{ij\ell m}(t)\, P_\ell(\xi)\, P_m(\eta) \quad \text{for}$$

$$-1 \leq \xi, \eta \leq 1, \quad (18)$$

where $\hat{U}_{ij\ell m}(t)$ is given by

$$\hat{U}_{ij\ell m}(t) = \frac{(2\ell + 1)(2m + 1)}{4} \int_{-1}^{1} \int_{-1}^{1} U(\xi, \eta, t)\, P_\ell(\xi)$$

$$P_m(\eta)\, d\xi\, d\eta. \quad (19)$$

The weak formulation (16) is simplified by mapping the integrals onto $\tilde{\Omega}_{ij}$ using the transformation (17) and exploiting the properties of Legendre polynomials. The computational formulation is analogous to (11) and is written as follows:

$$\frac{d}{dt} \hat{U}_{ij\ell m}(t) = \frac{(2\ell + 1)(2m + 1)}{2\Delta x_i \Delta y_j} [I_Q + I_F + I_G], \quad (20)$$

where

$$I_Q = \int_{-1}^{1} \int_{-1}^{1} \left[\Delta y_j F[U(\xi, \eta, t)] \frac{\partial P_\ell(\xi)}{\partial \xi} P_m(\eta) \right.$$

$$\left. + \Delta x_i G[U(\xi, \eta, t)] P_\ell(\xi) \frac{\partial P_m(\eta)}{\partial \eta} \right] d\xi d\eta, \quad (21)$$

$$I_F = -\Delta y_j \int_{-1}^{1} \{\hat{F}[U(1, \eta)](t) - (-1)^\ell$$

$$\hat{F}[U(-1, \eta)](t)\} P_m(\eta)\, d\eta, \quad (22)$$

$$I_G = -\Delta x_i \int_{-1}^{1} \{\hat{G}[U(\xi, 1)](t) - (-1)^m$$

$$\hat{G}[U(\xi, -1)](t)\} P_\ell(\xi)\, d\xi. \quad (23)$$

Here, I_Q is the surface integral for the gradient term in (16), and I_F and I_G are boundary flux integrals in the x and y directions, respectively. These integrals are evaluated with a high-order accurate GLL quadrature rule; $\hat{F}$ and $\hat{G}$ are numerical fluxes (4) at the element interfaces. The ODE (20) is then solved by the procedure given in (13).

According to the analysis performed in Cockburn and Shu (1989), a quadrature rule for the edges of the element (flux integrals I_F and I_G), must be exact for polynomials of degree $2k + 1$, and the quadrature rule for the interior of the elements (I_Q) must be exact for polynomials of degree $2k$, if the highest degree of the Legendre polynomials in $\mathcal{B}$ is k. Because we use a GLL grid, it is convenient to use a GLL quadrature rule that is accurate for polynomials of degree $2k + 1$. This choice may not be as efficient as using Gauss–Legendre quadrature, where fewer quadrature points are needed to gain the same accuracy. However, the choice of Gauss–Legendre quadrature needs further scaling to fit into the reference element $[-1, +1] \otimes [-1, +1]$.

4. Application of the DG method on the cubed sphere

a. Cubed-sphere geometry

Sadourny (1972) originally introduced the quasi-uniform spherical grid to avoid the pole problems associated with conventional spherical grids. However, his contribution remained dormant for many years. In the past decade, the approach has been recognized as a powerful tool for global modeling (e.g., Ronchi et al.

1996; Rančić et al. 1996; McGregor 1996; Taylor et al. 1997). The notational conventions adopted by Rančić et al. (1996) and Thomas and Loft (2002) are employed to elucidate the cubed-sphere geometry (also known as the spherical cube, or expanded spherical cube). Here a sphere is decomposed into six identical regions, obtained by central projection of the faces of the inscribed cube onto the spherical surface. Each of the six local coordinate systems is free of singularites and employs the identical metric terms, thus creating a nonorthogonal curvilinear coordinate system on the sphere.

Consider the cube with sides of length $2a$ inscribed into a sphere of radius R such that the eight vertices of the cube exactly touch the sphere and $a = R/\sqrt{3}$. The cube is oriented in such a way that the 3D absolute Cartesian coordinate axes (X, Y, Z) are normal to the faces (see Fig. 1). Let (x, y) be the local Cartesian coordinates centered on the surface of the cube face such that $x, y \in [-a, +a]$ and (λ, θ) denotes spherical longitude and latitude coordinates, where

$$X = R \cos\lambda \, \cos\theta$$
$$Y = R \sin\lambda \, \cos\theta$$
$$Z = R \sin\theta$$

(24)

Lateral faces (panels) are identified by $P_n, n = 1, \ldots,$ 4, whereas the top and bottom faces are P_5 and P_6, respectively. Figure 2 illustrates schematically the orientation of the different cube faces and their local con-

FIG. 2. Schematic for the relative positions of six cube faces (panels P_n, $n = 1, \ldots, 6$), and their local connectivity.

nectivity. Let P_1 be the cube face with X as the normal direction and consider the first positive octant of the cube; then from Fig. 1 the following trigonometric relations hold:

$$\sin\theta = \frac{Z}{R} = Z, \tan\lambda = \frac{x}{a} = \frac{Y}{X} \quad \text{and} \quad \frac{Y}{Z} = \frac{x}{y},$$

$$r^2 = a^2 + x^2 + y^2.$$

From the above relations and (24), it is clear that

$$x = a \tan\lambda$$
$$y = a \tan\theta \, \sec\lambda$$

(25)

Equation (25) is the basic gnomonic transformation between the cube and its circumscribing sphere. The transformation laws on the remaining five faces are obtained by rotating the sphere. Further details may be found in appendix A.

It is now possible to compute the metric tensor and corresponding quantities using (25). Let (x, y) be the local Cartesian coordinates defined on a cube face such that $\mathbf{r}$ is the corresponding position vector on the surface of the sphere. The covariant base vectors of the transformation between cube face and spherical surface are defined by

$$\mathbf{a}_1 = \frac{\partial \mathbf{r}}{\partial x}, \mathbf{a}_2 = \frac{\partial \mathbf{r}}{\partial y}.$$

(26)

The components of a covariant vector are given by $u_1 = \mathbf{v} \cdot \mathbf{a}_1$, $u_2 = \mathbf{v} \cdot \mathbf{a}_2$, and the corresponding contravariant components are expressed as

FIG. 1. Schematic illustration of gnomonic (central) mapping between sphere with radius R and the inscribed cube with side of length $2a$. Note that only one-eighth of the cube is shown in the figure. A point on the cube face P_1 is marked by a solid square with local Cartesian coordinates (x, y), and the corresponding point on the sphere is marked by a solid circle with absolute Cartesian coordinates (X, Y, Z). The spherical polar coordinates, longitude and latitude, are λ and θ, respectively; r is the radial distance from the center to any point (x, y) on the cube face.

$$\mathbf{v} = u^1 \mathbf{a}_1 + u^2 \mathbf{a}_2. \tag{27}$$

The metric tensor for the above transformation is

$$g_{ij} \equiv a_i \cdot a_j = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} = \frac{R^2}{r^4} \begin{bmatrix} a^2 + y^2 & -xy \\ -xy & a^2 + x^2 \end{bmatrix}, \tag{28}$$

where $i, j \in \{1, 2\}$. The covariant and contravariant vectors are related through the metric tensor g_{ij}, such that $u_i = g_{ij} u^j$ and $u^i = g^{ij} u_j$, where

$$g^{ij} = (g_{ij})^{-1} = \begin{bmatrix} g^{11} & g^{12} \\ g^{21} & g^{22} \end{bmatrix} = \frac{1}{g} \begin{bmatrix} g_{22} & -g_{12} \\ -g_{21} & g_{11} \end{bmatrix} \tag{29}$$

and $g = \det(g_{ij})$. The Jacobian of the transformation is $\sqrt{g} = R^2 a / r^3$. Note that the identical metric tensor g_{ij} is used on all the cube faces, irrespective of the local transformations (25). Appendix B derives the metric tensor (28) and its properties.

Two different central projections are considered in order to formulate (14) on the cubed sphere, employing the local Cartesian coordinates (x, y) (Sadourny 1972) and the equiangular (central) coordinates (α, β) (Ronchi et al. 1996; Thomas and Loft 2002). In addition, Rančić et al. (1996) developed an orthogonal curvilinear coordinate system based on conformal mapping.

b. Conservative transport on the cubed sphere

The continuity equation in flux form, on the sphere, is written as follows:

$$\frac{\partial \phi}{\partial t} + \operatorname{div}(\phi \mathbf{v}) = 0, \tag{30}$$

where ϕ is the advecting field and $\mathbf{v} = \mathbf{v}(\lambda, \theta)$ is the horizontal wind vector. Here, (30) is a particular case of the general form (14). In curvilinear coordinates, (30) can be written as (Sadourny 1972)

$$\frac{\partial}{\partial t}\left(\sqrt{g}\phi\right) + \frac{\partial}{\partial x}\left(\sqrt{g}u^1\phi\right) + \frac{\partial}{\partial y}\left(\sqrt{g}u^2\phi\right) = 0, \tag{31}$$

where (x, y) are considered to be contravariant independent variables. Usually, the velocity field $\mathbf{v} = (u, v)$ is given on the surface of the sphere; however, contravariant components of the velocity field are needed in (31).

The spherical velocity (u, v) is written in terms of (u^1, u^2) using (27) as follows:

$$\mathbf{A}\begin{bmatrix} u^1 \\ u^2 \end{bmatrix} = \begin{bmatrix} u \\ v \end{bmatrix}, \mathbf{A}^{-1}\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} u^1 \\ u^2 \end{bmatrix}, \tag{32}$$

where $\mathbf{A}$ is given by $g_{ij} = \mathbf{A}^T \mathbf{A}$, and $\mathbf{A}$ and $\mathbf{A}^{-1}$ are interpreted as the "cube-to-sphere·" and "sphere-to-cube" transformation matrices. However, the elements of the matrix $\mathbf{A}$ depend on the central projection method, local transformation laws, and local coordinates. Appendix B provides further details of the local transformations between cube faces and the sphere as well as the conversion of vector quantities to contravariant and covariant components.

1) EQUIDISTANT CENTRAL PROJECTION

Equation (31) can be discretized in the local Cartesian space on each cube face, where the independent variables are $x, y \in [-a, +a]$. For an equidistant projection, $\Delta x = \Delta y$ is constant and the metric tensor is given by (28). In this case, the transformation matrix $\mathbf{A}$ in (32) is defined to be

$$\mathbf{A} = \begin{bmatrix} R \cos\theta\lambda_x & R \cos\theta\lambda_y \\ R\theta_x & R\theta_y \end{bmatrix}. \tag{33}$$

Note that the same $\mathbf{A}$ is employed on each of the lateral faces $P_n, n = 1, \ldots, 4$. However, it is different on faces P_5 and P_6. Expressions for the elements of the matrix $\mathbf{A}$ are given in appendix B.

2) EQUIANGULAR CENTRAL PROJECTION

An alternative formulation of (31) is based on an equiangular projection employing the central angles (α, β) as independent variables rather than the local Cartesian coordinates (x, y). These two coordinate systems are related through

$$x = a \tan\alpha, y = a \tan\beta \quad \alpha, \beta \in [-\pi/4, \pi/4]. \tag{34}$$

In this case x and y in (31) are replaced by α and β, and the metric tensor of the transformation is

$$g_{ij} = \frac{R^2}{\rho^4 \cos^2\alpha \cos^2\beta} \begin{bmatrix} 1 + \tan^2\alpha & -\tan\alpha \tan\beta \\ -\tan\alpha \tan\beta & 1 + \tan^2\beta \end{bmatrix}, \tag{35}$$

where $\rho^2 = 1 + \tan^2 \alpha + \tan^2 \beta$. The transformation matrix and Jacobian are detailed in appendix C. Rančić et al. (1996) showed that the equiangular projection results in a more uniformly spaced grid compared to the equidistant projection and is more suitable for finite-difference approximations.

In addition to the equidistant and equiangular projections, we also consider a combination of these two projection methods. Here, the equiangular values of (α, β) are employed through (34) to obtain the independent variables (x, y). The metric terms for this case are the same as those considered for the equidistant formulation; however, $\Delta x \neq \Delta y$.

3) TREATMENT OF VECTORS ALONG THE CUBE-FACE EDGES

The treatment of vector quantities such as flux terms requires special attention along cube edges, because the

contiguous faces of the cube employ different local co-ordinate systems. Consider two adjoining elements separated by a cube edge and belonging to the cube faces P_n and P_l, such that $n, l \in \{1, \ldots, 6\}$ and $n \neq l$. Let (F_n^-, G_n^-) and (F_l^+, G_l^+) be the left and right vectors located at a common point (a GLL point) but having two different local coordinate systems. In the case of flux vectors, the notation $F^\pm = F(U^\pm)$ and $G^\pm = G(U^\pm)$ is consistent with section 3. To compute the flux using (5) in the x direction, on the edge of P_n, both the *left* and *right* components F_n^- and F_n^+ are required. The right component F_n^+ is computed from (F_l^+, G_l^+) by the procedure described below.

First, transform (F_l^+, G_l^+) defined on P_l into the corresponding spherical components (F_s^+, G_s^+) by applying the matrix A_l. Next, transform (F_s^+, G_s^+) into (F_n^+, G_n^+) defined on P_n by applying the matrix $\mathbf{A}_n^{-1}$. The dual transformations based on (32) are summarized below:

$$\mathbf{A}_l \begin{bmatrix} F_l^+ \\ G_l^+ \end{bmatrix} = \begin{bmatrix} F_s^+ \\ G_s^+ \end{bmatrix}, \mathbf{A}_n^{-1} \begin{bmatrix} F_s^+ \\ G_s^+ \end{bmatrix} = \begin{bmatrix} F_n^+ \\ G_n^+ \end{bmatrix}$$

$$\Rightarrow \mathbf{A}_n^{-1} \mathbf{A}_l \begin{bmatrix} F_l^+ \\ G_l^+ \end{bmatrix} = \begin{bmatrix} F_n^+ \\ G_n^+ \end{bmatrix}. \tag{36}$$

The required numerical flux along the x direction, $\hat{F}(U^-, U^+)$, is computed according to (5). This procedure may be extended to any cube-face edge for computing vector quantities. Metric terms, transformation matrices, and dual transformation matrices $\mathbf{A}_n^{-1}\mathbf{A}_l$ may be precomputed.

5. Numerical experiments

The flux form continuity Eq. (31) can be written in the computational form considered in section 3, with $U = \sqrt{g}\phi$ and the fluxes $F(U) = \sqrt{g}u^1\phi$ and $G(U) = \sqrt{g}u^2\phi$. Independent variables (contravariant) are either (x, y) or (α, β). Two standard advection tests, solid-body rotation and deformational flow, were performed to evaluate the DG scheme.

a. Solid-body rotation of a cosine bell

The "cosine bell" problem proposed by Williamson et al. (1992) is often employed to test global advection schemes. The initial scalar field is defined as follows:

$$\phi(\lambda, \theta) = \begin{cases} (h_0/2)[1 + \cos(\pi r_d/r_0)] & \text{if } r_d < r_0 \\ 0 & \text{if } r_d \geq r_0, \end{cases} \tag{37}$$

where r_d is the great-circle distance between (λ, θ) and the bell center. The bell of radius $r_0 = R/3$ is initially centered at $(3\pi/2, 0)$ corresponding to the central point of P_4 on the cube (see Fig. 2). The maximum height of the cosine bell is $h_0 = 1000$ m; $R = 6.37122 \times 10^6$ m is the earth's radius; and velocity components of the advecting wind field are

$$u = u_0(\cos\alpha_0 \cos\theta + \sin\alpha_0 \cos\lambda \sin\theta),$$

$$v = -u_0 \sin\alpha_0 \sin\lambda,$$

where $u_0 = 2\pi R/(12 \text{ days})$, and α_0 is the angle between the axis of solid-body rotation and the polar axis of the spherical coordinate system (Williamson et al. 1992). When the value of α_0 is equal to zero or $\pi/2$, the flow direction is along the equator or in the north–south (meridional) direction, respectively. For the cubed sphere, flow along the northeast direction ($\alpha_0 = \pi/4$) is more challenging, because the cosine-bell pattern passes over four vertices and two edges of the cube to complete revolution.

b. Deformational flow

For the deformational flow test, an idealized cyclogenesis problem (Doswell 1984) is simulated in spherical geometry (Nair et al. 1999). The test is also described in Nair and Machenhauer (2002) and Nair et al. (2003). Two steady vortices are generated on the sphere such that the vortex centers are located near the two vertices of the cube.

The normalized tangential velocity of the vortex field on a unit sphere is given by

$$V_t = \frac{3\sqrt{2}}{2} \text{sech}^2(\rho') \tanh(\rho'),$$

where $\rho' = \rho_0' \cos\theta'$ is the radial distance of the vortex. The exact solution at time t is

$$\phi(\lambda', \theta', t) = 1 - \tanh\left[\frac{\rho'}{\gamma} \sin(\lambda' - \omega' t)\right], \tag{38}$$

where (λ', θ') are the rotated spherical coordinates with poles at the vortex centers, and γ is a parameter defining the characteristic width (Nair et al. 1999). The angular velocity ω' is specified as

$$\omega'(\theta') = \begin{cases} 0 & \text{if } \rho' = 0 \\ V_t/\rho' & \text{if } \rho' \neq 0. \end{cases} \tag{39}$$

Velocity components on the unit sphere are defined as

$$u = \omega'(\sin\theta_0 \cos\theta + \cos\theta_0 \cos(\lambda - \lambda_0) \sin\theta),$$

$$v = \omega' \cos\theta_0 \sin(\lambda - \lambda_0),$$

where (θ_0, λ_0) is the north pole of the rotated spherical coordinate system (λ', θ'). For the present study, the parameters chosen are $(\lambda_0, \theta_0) = (\pi - 0.8, \pi/4.8)$, $\rho_0' = 3$, and $\gamma = 5$. Two symmetric vortices are generated near two diametrically opposite cubed-sphere vertices. These vortex centers are located approximately at the top-right corner of P_4 and the bottom-right corner of P_2

in Fig. 2. Thus, the deforming fields around the vortex centers are supported by three contiguous local domains.

c. Results

The above tests were performed using grid resolutions defined by $M \times N_g \times N_g$, where M is the number of elements with $N_g \times N_g$ GLL points in an element. However, results are only reported for two grid resolutions, a "low order" grid with $M = 2400$, $N_g = 4$, and a "high order" grid with $M = 96$, $N_g = 16$. The latter grid ($96 \times 16 \times 16$) was used for comparing our results with the SE method (Thomas and Loft 2002).

First consider advection of the cosine bell along the northeast direction on the low-order grid ($2400 \times 4 \times 4$) in order to compare equidistant (x, y) (case A) and equiangular (α, β) (case C) central projections. In addition, a combination of these two projections (case B)
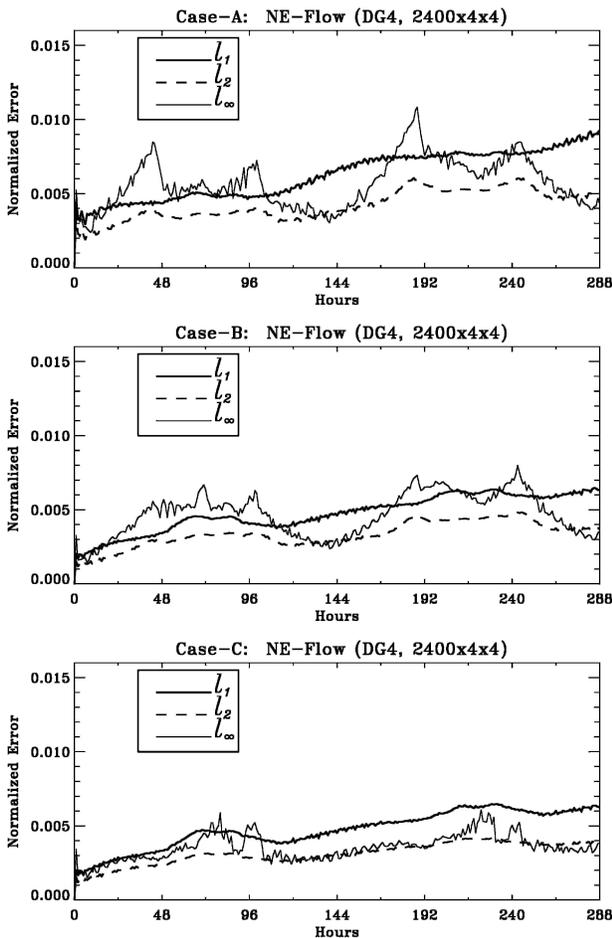
FIG. 4. Results on an orthographic projection for solid-body rotation of a cosine bell after one revolution (12 days) along the northeast direction. Contours are plotted with a uniform interval of 100 m. (right) The position of the cosine bell when its center reaches a corner point of the cubed sphere. Note that every tile shown on the cubed sphere consists of 4 elements, and 2400 elements span the surface of the sphere.

was examined. Here, the independent variables (x, y) are generated with the equiangular values of α and β in the interval $[-\pi/4, \pi/4]$, employing (34).

Figure 3 illustrates the time evolution of the standard ℓ_1, ℓ_2 and ℓ_∞ relative errors during one complete revolution (12 days) of the cosine bell in the northeast direction (Williamson et al. 1992). A low-order grid ($2400 \times 4 \times 4$) with third-order explicit Runge–Kutta integrator (13) is applied for 2880 time steps ($\Delta t = 6$ min) to complete one revolution. The top-to-bottom panels in Fig. 3 are the error plots for the central projections in cases A, B, and C. It is clear that the equiangular projection (case C) results in smaller errors. In particular, the ℓ_∞ error is significantly reduced. Case B exhibits smaller error norms than case A. Equiangular projection (case C) results in a more uniform distribution of GLL points on the cubed sphere (Rančić et al.1996) and error metrics in case C are far superior. Hereafter, experiments using the equiangular projection are reported.

The left panel in Fig. 4 shows the cosine bell after one revolution in the northeast direction after returning to the initial position. Note in this case that the data were bilinearly interpolated from the source grid consisting of six identical subdomains onto a 128×65 latitude–longitude grid, for orthographic visualization. Contours are plotted with uniform interval of 100 m. The numerical solution with DG is visibly indistinguishable from the analytic solution (plotted as dashed contours). The right panel in Fig. 4 displays the numerical solution centered at the first vertex away from the initial position. The cosine bell smoothly crosses over the corner point and does not exhibit any shocks or discontinuities while passing over cube-face edges.

Figure 5 is a 3D plot of the cosine bell after one complete revolution along the northeast, north–south ($\alpha_0 = \pi/2$) and equatorial ($\alpha_0 = 0$) directions. The reference (initial) solution is shown (top-left panel). Here, the DG solution is plotted at the corner points of ele-

FIG. 3. Time traces of normalized ℓ_1, ℓ_2 and ℓ_∞ errors for the cosine-bell advection problem in the northeast direction, with different central projections from cube to sphere. (top) The errors plots for the equidistant projection (case A) and the equivalent for the equiangular projections for (middle) case B and (bottom) case C.
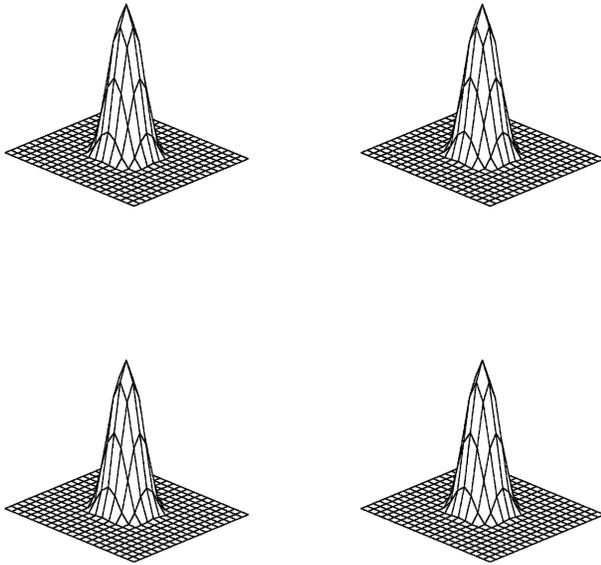
FIG. 5. Results for solid-body rotation of a cosine bell after one revolution (12 days) along the (top right) northeast, (bottom left) north–south, and (bottom right) equatorial directions. (top left) The initial (reference) solution. Corner values of the elements on the cube face P_4 (see Fig. 2) are used for plotting the height field, with initial maximum value 1000 units.

ments extracted from the panel P_4, where the cosine bell is initially centered. Note that the numerical solution is plotted without any postprocessing interpolations. Figure 6 shows the difference field (DG versus exact solution) for the three different flow orientations in Fig. 5. The maximum difference (nonnormalized) is less than 1%. The basic structure of the solution (phase and amplitude) does not change depending on the flow orientation (no directional bias).

Next, the DG results are compared with the SE method described in Thomas and Loft (2002) for the 96 $\times$ 16 $\times$ 16 grid, using equiangular central projection. The duration of the integration is 12 days (one revolution around the sphere along the northeast direction). Time integration for the DG method was performed using a third-order TVD Runge–Kutta scheme with a time step $\Delta t = 30$ s. Neither limiters nor spatial filters were employed for the DG integration. For the SE model (Thomas and Loft 2002) the same time-integration scheme was used; however, even with a TVD time integrator a filter was found to be essential to stabilize the integration. A Boyd–Vandeven filter was applied every 20 min for the SE method. Figure 7a illustrates the time evolution of ℓ_1, ℓ_2 and ℓ_∞ errors for

$\rightarrow$
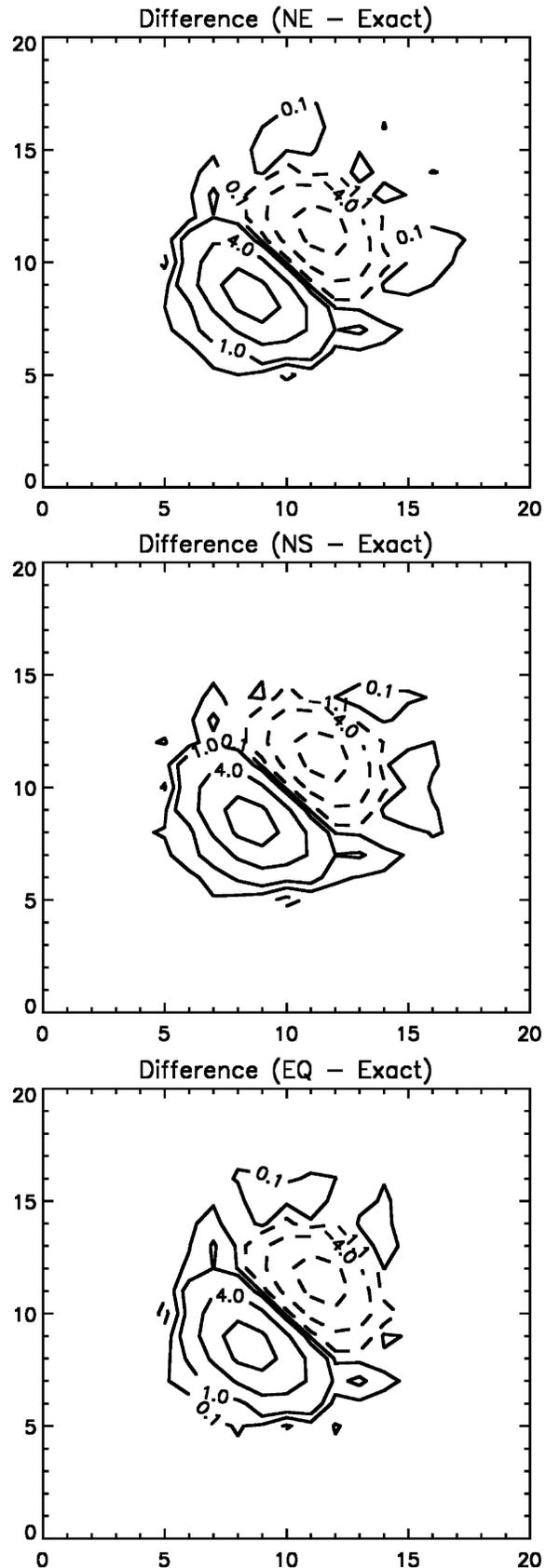
FIG. 6. The height difference field (numerical minus exact solution) of the cosine bell as shown in Fig. 5, when the flow orientation is along the (bottom) equatorial, (middle) north–south, and (top) northeast directions.
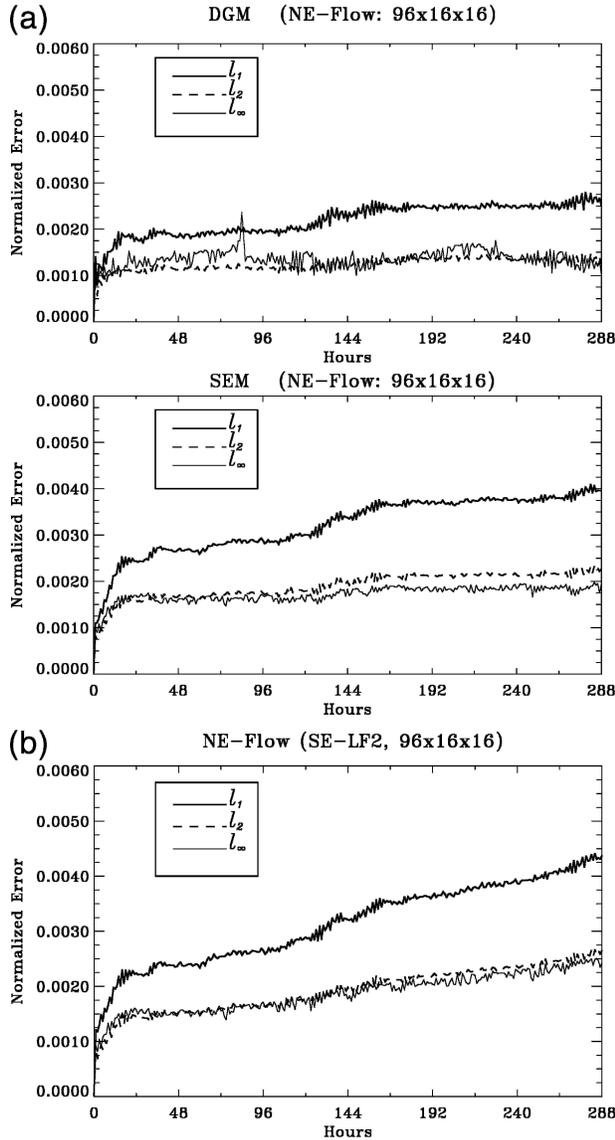
FIG. 7. (a) Time traces of normalized ℓ_1, ℓ_2 and ℓ_∞ errors of the cosine bell advection problem along the northeast direction. (top) The results with a discontinuous Galerkin method (DGM), and (bottom) the results with a spectral element method (SEM). For both methods, a third-order TVD Runge–Kutta method is used, and the domain consists of 96 elements, each with 16×16 Gauss–Lobatto–Legendre points. (b) Same as in (a), but for the SEM. Time integration for this case is a second-order leapfrog scheme combined with a Boyd–Vandeven filter.

al. (1997), who used a combination of equidistant and equiangular projections. This again shows the advantage of using an equiangular projection.

Convergence of the DG scheme is displayed in Figs. 8a and 8b, for the cosine-bell problem. We have used h-refinement (i.e., keeping the degree of the polynomial fixed and increasing the number of elements) and p-refinement (i.e., increasing the degree of the polynomial for a fixed total number of elements) convergence tests, as is typically used in high-order methods. Total number of elements on the cubed sphere can be expressed as $M = 6 \, N_e \times N_e$, where N_e is the number of elements in x or y direction on any cube face. Figure 8a
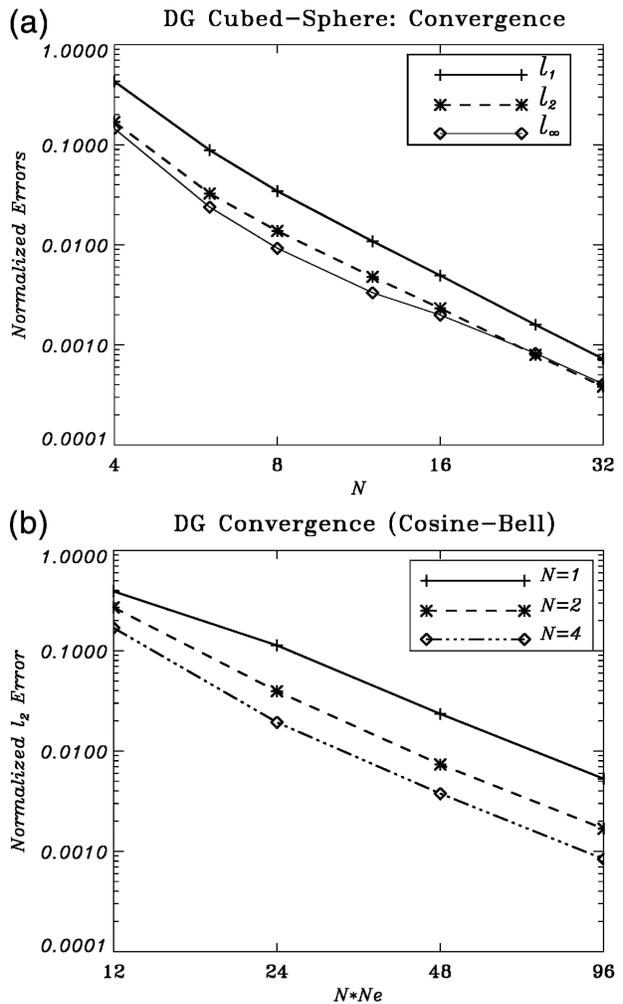
FIG. 8. (a) Convergence of DG scheme for the cosine-bell problem with different values of the degree N of Legendre polynomials. Normalized errors (ℓ_1, ℓ_2 and ℓ_∞) are computed after one complete revolution of the cosine bell along the northeast direction of the cubed sphere. (b) Convergence of DG scheme for the cosine-bell problem as function of number of elements ($N \times N_e$), for the Legendre polynomials of degree $N = 1$, 2, and 4. Here, N_e is the number of elements used in the x of y direction on any cube face, and the total number of elements on the cubed sphere is given by $M = 6 \times N_e^2$.

the DG (top panel) and SE (bottom panel) discretizations. The DG solution exhibits smaller error growth rates when compared to the SE solutions. However, the ℓ_∞ error for the DG scheme appears to be more oscillatory than the SE scheme. Figure 7b shows SE results with an explicit second-order leapfrog scheme ($\Delta t = 30$ s), and a Boyd–Vandeven filter was applied every 20 min. The error norms for this case are much smaller than the results obtained in a similar study by Taylor et

shows a *p*-refinement convergence for $N_e = 3$ (or $M = 54$) and the degree of the Legendre polynomial N varies from 4 to 32. The ℓ_2 error shown in Fig. 8a compares that of Fig. 6 (for the cosine-bell problem) in Giraldo et al. (2002). A strict comparison with Giraldo et al. (2002) is not possible because of the disparity of the grid systems used. However, with a fewer number of elements ($M = 54$), we observe the same error levels reported in Giraldo et al. (2002) with $M = 60$. Figure 8b shows *h*-refinement convergence for $N = 1, 2,$ and 4, and number of elements as a function of $N \times N_e$. For example, when $N = 4$, $N_e = 3, 6, 12,$ and 24 and corresponding total number of elements on the cubed sphere are 54, 216, 864, and 3456, respectively.

Normalized global mass (integral invariant) was computed as a function of time using Eq. (135) of Williamson et al. (1992), and the result is shown in Fig. 9. The normalized mass is conserved to the machine precision as expected and is found to be independent of the particular grid resolution used (i.e., M or N).

Deformational flow results are presented in Fig. 10. The top-left and right panels show the initial and exact solutions at 6 days. The numerical solution on a 2400 × 4 × 4 grid is shown in the bottom-left panel. For the TVD Runge–Kutta integrator (13), 1440 × Δt = 6 min time steps (equivalent to 6 days) were used to simulate the vortices. For the vortices in Fig. 10, contours vary from 0.5 to 1.5, with a central value 1.0 displayed as a thick line. The numerical solution is visibly identical to the exact solution. The evolution of the vortices (at diametrically opposite vertices of the cubed sphere) have been well simulated by the DG scheme, even at the corners (consisting of three subdomains). The bottom-right panel shows the difference field with contour values 1×10^{-6} (thick contours) and -1×10^{-6} (thin contours) and do not exhibit any bias along the edges. Figures 11a and 11b, respectively, show the *p*- and *h*-type convergence for the vortex problem. Global mass
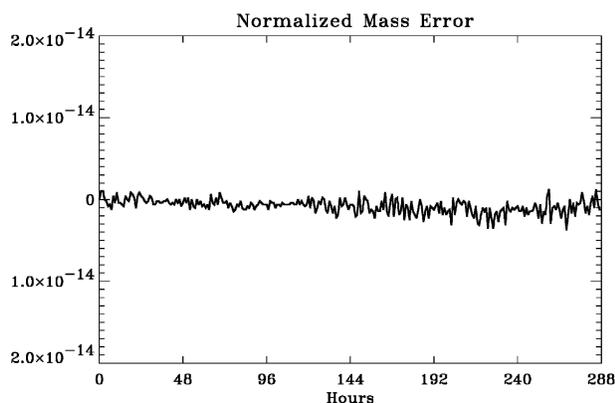
FIG. 10. (top left) The initial flow field, and (top right) the exact solution (at t = 6 days). The vortex center is located near a vertex of the cubed sphere. (bottom left) Numerical solutions are produced by the DG scheme (see text). (bottom right) The difference field (numerical minus exact). The contour values are $1 - 10^{-6}$ (thick) and -1×10^{-6} (thin).

error for all the experiments was computed and found to be on the order of machine precision (plots are not shown) and independent of the central projection and grid type (resolution).

6. Summary and conclusions

We have proposed a conservative transport scheme based on the discontinuous Galerkin (DG) method on the cubed sphere. Two different central projection methods, equidistant and equiangular, were employed to map between the inscribed cube and the sphere. These mappings divide the spherical surface into six identical subdomains and are free from singularities. Two standard advection tests, solid-body rotation and deformational flow, were performed to evaluate the DG scheme. A third-order explicit total variation diminishing Runge–Kutta scheme was employed for time integration without application of a filter or limiter to control spurious oscillations. A cosine bell was utilized to evaluate the performance of the DG scheme for advection. The standard ℓ_1, ℓ_2 and ℓ_∞ relative error metrics are found to be smaller for the equiangular projection compared to the equidistant projection. The cosine bell passes smoothly over the vertices and edges, while maintaining high-order accuracy. Our results were compared against a standard spectral element (SE) method employing equiangular central projection

FIG. 9. Time traces of total normalized mass error for the cosine-bell problem for one complete revolution. Note that the total mass is conserved to machine precision (double-precision arithmetic) and is independent of the number of elements used and polynomial degree.

(a)

(b)
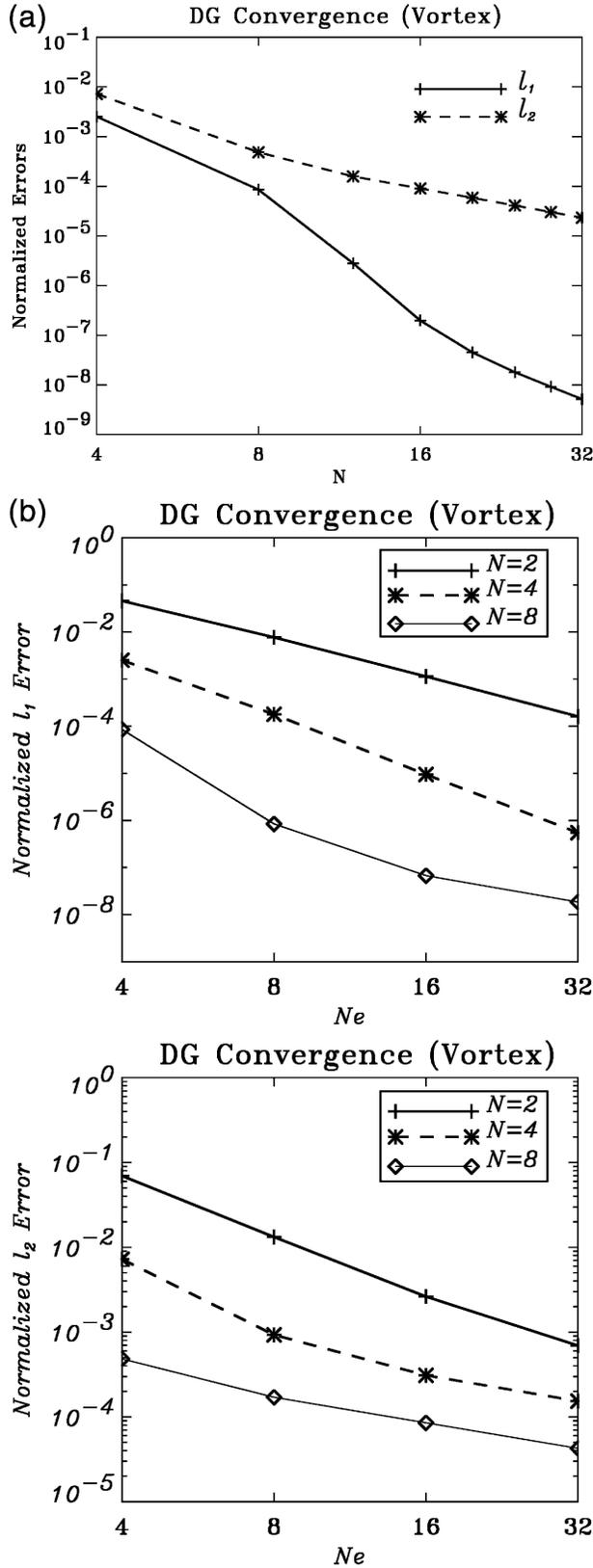
(Thomas and Loft 2002). Both simulations employed a $96 \times 16 \times 16$ grid and a third-order TVD Runge–Kutta time-integration scheme. However, a filter is necessary for the SE method to control spurious oscillations. Time series of the standard error metrics grew more slowly in the case of the DG scheme. In particular, the error metrics for the SE method (Thomas and Loft 2002) employing a second-order leapfrog scheme were found to be significantly smaller than those reported by Taylor et al. (1997) for the same experiment. This again shows that on the cubed sphere with the same number of grid points, an equiangular projection is more accurate than an equidistant projection.

A deformational flow test that generates vortices near two diametrically opposite vertices of the cubed sphere was used to evaluate the robustness of the DG scheme. Numerical solutions in this case were found to be very accurate, including near the vortex centers and edges of the cube. The numerical solution on the cubed sphere demonstrates excellent convergence properties for the scalar advection problem and achieves the accuracy using fewer degrees of freedom than the recent study by Giraldo et al. (2002). A flux form shallow water model on the cubed sphere based on the DG method is under development. Implementation of monotonic limiters in the DG scheme is also under development, and we plan to report these results in the near future.

APPENDIX A

Transformation Laws for the Six Cube Faces

Let (λ', θ') be a rotated spherical coordinate system having the north pole at (λ_0, θ_0) with respect to the regular (λ, θ) coordinate system. Local Cartesian coordinates (x, y) defined on the six faces of the cube can be expressed in terms of (λ, θ) by rotating the sphere for different values of (λ_0, θ_0). The following spherical trigonometric equations (Nair et al. 1999) may be employed for projecting each face of the cube:

$$\tan\lambda' = \frac{\sin(\lambda - \lambda_0)}{\sin\theta_0 \cos(\lambda - \lambda_0) - \cos\theta_0 \tan\theta}, \quad \text{(A1)}$$

$$\sin\theta' = \sin\theta \sin\theta_0 + \cos\theta_0 \cos(\lambda - \lambda_0). \quad \text{(A2)}$$

Fig. 11. (a) Same as in Fig. 8a but for the vortex (deformational flow) problem. Normalized errors ℓ_1 and ℓ_2 are used for the convergence study. (b) Same as in Fig. 8b but for the vortex (deformational flow) problem. (top) The convergence of the DG scheme for normalized ℓ_1 error, and (bottom) the ℓ_2 error.

- On P_1: set $(\lambda_0, \theta_0) = (0, \pi/2)$; (A1) and (A2) imply that $\lambda' = \lambda$, $\theta' = \theta$. Now (x, y) are defined as

$$x = a \tan\lambda' = a \tan\lambda,$$
$$y = a \tan\theta' \sec\lambda' = a \tan\theta \sec\lambda \Big\}. \quad (A3)$$

The absolute Cartesian coordinates and (x, y) are related through

$$(X, Y, Z) = \frac{R}{r}(a, x, y), (x, y) = a\left(\frac{Y}{X}, \frac{Z}{X}\right).$$

- On P_2: set $(\lambda_0, \theta_0) = (\pi/2, \pi/2) \Rightarrow \lambda' = \lambda - \pi/2$, $\theta' = \theta$,

$$x = a \tan(\lambda - \pi/2) = -a \cot\lambda$$
$$y = a \tan\theta \sec(\lambda - \pi/2) = a \tan\theta \csc\lambda \Big\},$$

$$(A4)$$

$$(X, Y, Z) = \frac{R}{r}(-x, a, y), (x, y) = \left(\frac{-X}{Y}, \frac{Z}{Y}\right).$$

- On P_3: set $(\lambda_0, \theta_0) = (\pi, \pi/2) \Rightarrow \lambda' = \lambda - \pi$, $\theta' = \theta$,

$$x = a \tan(\lambda - \pi) = a \tan\lambda,$$
$$y = a \tan\theta \sec(\lambda - \pi) = -a \tan\theta \sec\lambda \Big\},$$

$$(A5)$$

$$(X, Y, Z) = \frac{R}{r}(-a, -x, y), (x, y) = a\left(\frac{Y}{X}, \frac{-Z}{X}\right).$$

- On P_4: set $(\lambda_0, \theta_0) = (3\pi/2, \pi/2) \Rightarrow \lambda' = \lambda - 3\pi/2$, $\theta' = \theta$,

$$x = a \tan(\lambda - 3\pi/2) = -a \cot\lambda,$$
$$y = a \tan\theta \sec(\lambda - 3\pi/2) = -a \tan\theta \csc\lambda \Big\},$$

$$(A6)$$

$$(X, Y, Z) = \frac{R}{r}(x, -a, y), (x, y) = a\left(\frac{-X}{Y}, \frac{-Z}{Y}\right).$$

- On P_5 (top panel): set $(\lambda_0, \theta_0) = (\pi, 0)$, from (A1) and (A2);

$$\tan\lambda' = \frac{\sin(\lambda - \pi)}{-\tan\theta} = \sin\lambda \cot\theta,$$

$$\sin\theta' = \cos\theta \cos(\lambda - \pi) = -\cos\lambda \cos\theta.$$

$$x = a \tan\lambda' = a \sin\lambda \cot\theta$$

$$y = a \tan\theta' \sec\lambda'$$

$$= a \frac{\sin\theta'}{\cos\theta'}\sqrt{1 + \tan^2\lambda'};$$

$$\left[\text{choose} \frac{\sqrt{1 + \tan^2\lambda'}}{\cos\theta'} = \frac{1}{+\sin\theta}\right]$$

$$= a \frac{-\cos\lambda \cos\theta}{\sin\theta} = -a \cos\lambda \cot\theta.$$

To summarize for P_5,

$$x = a \sin\lambda \cot\theta$$
$$y = -a \cos\lambda \cot\theta, \quad \theta \in [\pi/4, \pi/2] \Big\}. \quad (A7)$$

$$(X, Y, Z) = \frac{R}{r}(-y, x, a), \ (x, y) = a\left(\frac{Y}{Z}, \frac{-X}{Z}\right).$$

- On P_6 (bottom panel): set $(\lambda_0, \theta_0) = (2\pi, 0)$, from (A1) and (A2);

$$\tan\lambda' = \frac{\sin(\lambda - 2\pi)}{-\tan\theta} = -\sin\lambda \cot\theta,$$

$$\sin\theta' = \cos\theta \cos(\lambda - 2\pi) = \cos\lambda \cos\theta.$$

$$x = a \tan\lambda' = -a \sin\lambda \cot\theta$$

$$y = a \tan\theta' \sec\lambda'$$

$$= a \frac{\sin\theta'}{\cos\theta'}\sqrt{1 + \tan^2\lambda'},$$

$$\left[\text{choose} \frac{\sqrt{1 + \tan^2\lambda'}}{\cos\theta'} = \frac{1}{-\sin\theta}\right]$$

$$= a \frac{\cos\lambda \cos\theta}{-\sin\theta} = -a \cos\lambda \cot\theta.$$

To summarize for P_6,

$$x = -a \sin\lambda \cot\theta,$$
$$y = -a \cos\lambda \cot\theta, \quad \theta \in [-\pi/2, -\pi/4] \Big\},$$

$$(A8)$$

$$(X, Y, Z) = \frac{R}{r}(y, x, -a), \ (x, y) = a\left(\frac{-Y}{Z}, \frac{-X}{Z}\right).$$

APPENDIX B

Transformations of Vectors between Cube and Sphere Surfaces

Let (x, y) be arbitrary Cartesian independent variables on a cube face and $\mathbf{r}$ be the corresponding position vector on the sphere. In absolute Cartesian coordinates, denote $\mathbf{r} = (X, Y, Z)$ such that $d\mathbf{r} = (dX, dY, dZ)$. The basis vectors (26) for the transformations are $\mathbf{a}_1 = \mathbf{r}_x$, $\mathbf{a}_2 = \mathbf{r}_y$. On any cube face, the equations $(X, Y, Z) = [X(x, y), Y(x, y), Z(x, y)]$ in appendix A can be utilized to derive the explicit form of the metric tensor g_{ij}, $i, j \in \{1, 2\}$,

$$g_{ij} = \mathbf{a}_i \cdot \mathbf{a}_j = \begin{bmatrix} \mathbf{r}_x \cdot \mathbf{r}_x & \mathbf{r}_x \cdot \mathbf{r}_y \\ \mathbf{r}_y \cdot \mathbf{r}_x & \mathbf{r}_y \cdot \mathbf{r}_y \end{bmatrix}$$

$$= \frac{R^2}{r^4}\begin{bmatrix} a^2 + y^2 & -xy \\ -xy & a^2 + x^2 \end{bmatrix}, \quad (B1)$$

where $\mathbf{r}_x = (X_x, Y_x, Z_x)$ and $\mathbf{r}_y = (X_y, Y_y, Z_y)$ are obtained by direct differentiation. It can be verified

that tensor (B1) is the same for all local transformations.

Spherical coordinates (λ, θ) are emphasized in this development rather than the absolute Cartesian coordinates. For any horizontal vector **r** on the sphere, we have the following differential form

$$d\mathbf{r} = \mathbf{i}\,R\cos\theta\,d\lambda + \mathbf{j}\quad R\,d\theta, \qquad (B2)$$

where the vector is decomposed into the east (λ) and the north (θ) directions (Dutton 1986). Then basis vectors can be expressed as

$$\mathbf{a}_1 = \mathbf{r}_x = \mathbf{i}\,R\cos\theta\frac{d\lambda}{dx} + \mathbf{j}\,R\frac{d\theta}{dx}, \qquad (B3)$$

$$\mathbf{a}_2 = \mathbf{r}_y = \mathbf{i}\,R\cos\theta\frac{d\lambda}{dy} + \mathbf{j}\,R\frac{d\theta}{dy}. \qquad (B4)$$

The horizontal wind vector $\mathbf{v} = \mathbf{i}\,u + \mathbf{j}\,v$ is expressed in terms of contravariant components (27) as

$$\mathbf{v} = u^1\mathbf{a}_1 + u^2\mathbf{a}_2$$

$$\mathbf{i}\,u + \mathbf{j}\,v = \mathbf{i}\,R\left(u^1\cos\theta\frac{d\lambda}{dx} + u^2\cos\theta\frac{d\lambda}{dy}\right)$$

$$+ \mathbf{j}\,R\left(u^1\frac{d\theta}{dx} + u^2\frac{d\theta}{dy}\right),$$

or in the following matrix form:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \mathbf{A}\begin{bmatrix} u^1 \\ u^2 \end{bmatrix}, \qquad (B5)$$

where

$$\mathbf{A} = \begin{bmatrix} R\cos\theta\lambda_x & R\cos\theta\lambda_y \\ R\theta_x & R\theta_y \end{bmatrix}. \qquad (B6)$$

The basis vectors in (B3) and (B4) could be used to derive the tensor g_{ij} as given in (B1). The metric tensor g_{ij} may be expressed in terms of the matrix **A**:

$$g_{ij} = \begin{bmatrix} \mathbf{r}_x \cdot \mathbf{r}_x & \mathbf{r}_x \cdot \mathbf{r}_y \\ \mathbf{r}_y \cdot \mathbf{r}_x & \mathbf{r}_y \cdot \mathbf{r}_y \end{bmatrix}$$

$$= R^2\begin{bmatrix} \cos^2\theta\,\lambda_x^2 + \theta_x^2 & \cos^2\theta\,\lambda_x\lambda_y + \theta_x\theta_y \\ \cos^2\theta\,\lambda_x\lambda_y + \theta_x\theta_y & \cos^2\theta\,\lambda_y^2 + \theta_x^2 \end{bmatrix}$$

$$= \mathbf{A}^T\mathbf{A}. \qquad (B7)$$

The covariant and contravariant vectors are related through the metric tensor g_{ij} and its inverse g^{ij}, such that $u_i = g_{ij}u^j$ and $u^i = g^{ij}u_j$:

$$g^{ij} = (g_{ij})^{-1} = \begin{bmatrix} g^{11} & g^{12} \\ g^{21} & g^{22} \end{bmatrix} = \frac{1}{g}\begin{bmatrix} g_{22} & -g_{12} \\ -g_{21} & g_{11} \end{bmatrix}$$

$$= \mathbf{A}^{-1}\mathbf{A}^{-T}, \qquad (B8)$$

where $g = \det(g_{ij})$, and the Jacobian of the transformation is $\sqrt{g} = |\mathbf{a}_1 \times \mathbf{a}_2|$. To summarize, the transformations between covariant and contravariant components of **v** are

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}\begin{bmatrix} u^1 \\ u^2 \end{bmatrix}, \quad \text{and}$$

$$\begin{bmatrix} u^1 \\ u^2 \end{bmatrix} = \begin{bmatrix} g^{11} & g^{12} \\ g^{21} & g^{22} \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \qquad (B9)$$

The explicit form of the matrix **A** in (B6) is the same on lateral faces P_1 to P_4, but different for P_5 (top) and P_6 (bottom) panels.; **A** and $\mathbf{A}^{-1}$ can be derived as follows. On P_1 to P_4, using the basic form (25),

$$\lambda_x = \frac{1}{a}\cos^2\lambda, \quad \lambda_y = 0, \quad \theta_x = -\frac{1}{a}\sin\theta\sin\lambda\cos\theta\cos\lambda,$$

$$\theta_y = \frac{1}{a}\cos^2\theta\cos\lambda,$$

$$\mathbf{A} = \frac{R\cos\theta\cos\lambda}{a}\begin{bmatrix} \cos\lambda & 0 \\ -\sin\theta\sin\lambda & \cos\theta \end{bmatrix},$$

$$\mathbf{A}^{-1} = \frac{a\sec\theta\sec\lambda}{R}\begin{bmatrix} \sec\lambda & 0 \\ \tan\theta\tan\lambda & \sec\theta \end{bmatrix}. \qquad (B10)$$

On P_5, using (A7),

$$\lambda_x = \frac{1}{a}\cos\lambda\tan\theta, \quad \lambda_y = \frac{1}{a}\sin\lambda\tan\theta,$$

$$\theta_x = \frac{-1}{a}\sin\lambda\sin^2\theta, \quad \theta_y = \frac{1}{a}\cos\lambda\sin^2\theta,$$

$$\mathbf{A} = \frac{R\sin\theta}{a}\begin{bmatrix} \cos\lambda & \sin\lambda \\ -\sin\theta\sin\lambda & \sin\theta\cos\lambda \end{bmatrix},$$

$$\mathbf{A}^{-1} = \frac{a}{R\sin^2\theta}\begin{bmatrix} \sin\theta\cos\lambda & -\sin\lambda \\ \sin\theta\sin\lambda & \cos\lambda \end{bmatrix}. \qquad (B11)$$

On P_6, using (A8),

$$\lambda_x = \frac{-1}{a}\cos\lambda\tan\theta, \quad \lambda_y = \frac{1}{a}\sin\lambda\tan\theta,$$

$$\theta_x = \frac{1}{a}\sin\lambda\sin^2\theta, \quad \theta_y = \frac{1}{a}\cos\lambda\sin^2\theta,$$

$$\mathbf{A} = \frac{R\sin\theta}{a}\begin{bmatrix} -\cos\lambda & \sin\lambda \\ \sin\theta\sin\lambda & \sin\theta\cos\lambda \end{bmatrix},$$

$$\mathbf{A}^{-1} = \frac{a}{R\sin^2\theta}\begin{bmatrix} -\sin\theta\cos\lambda & \sin\lambda \\ \sin\theta\sin\lambda & \cos\lambda \end{bmatrix}. \qquad (B12)$$

APPENDIX C

Equiangular Formulation between Cube and Sphere Surfaces

An alternative formulation of (31) is possible when the central angles $\alpha, \beta \in [-\pi/4, \pi/4]$ are independent

variables instead of local Cartesian coordinates (x, y) on the cube faces. These central projections are related through

$$x = a \tan\alpha, \quad y = a \tan\beta. \tag{C1}$$

For an equiangular projection, the basis vectors $\mathbf{a}_1 = \mathbf{r}_\alpha$ and $\mathbf{a}_2 = \mathbf{r}_\beta$ may be written as

$$\mathbf{r}_\alpha = \frac{a}{\cos^2\alpha}\mathbf{r}_x, \quad \mathbf{r}_\beta = \frac{a}{\cos^2\beta}\mathbf{r}_y, \tag{C2}$$

where $\mathbf{r}_x$ and $\mathbf{r}_y$ are as defined in (B3) and (B4), respectively. Furthermore, the metric tensor can be derived as in the case of (B7),

$$
\begin{aligned}
g_{ij} &= \begin{bmatrix} \mathbf{r}_\alpha \cdot \mathbf{r}_\alpha & \mathbf{r}_\alpha \cdot \mathbf{r}_\beta \\ \mathbf{r}_\beta \cdot \mathbf{r}_\alpha & \mathbf{r}_\beta \cdot \mathbf{r}_\beta \end{bmatrix} \\
&= \frac{R^2}{\rho^4 \cos^2\alpha \cos^2\beta} \begin{bmatrix} 1 + \tan^2\alpha & -\tan\alpha \tan\beta \\ -\tan\alpha \tan\beta & 1 + \tan^2\beta \end{bmatrix} \\
&= \tilde{\mathbf{A}}^T \tilde{\mathbf{A}},
\end{aligned}
\tag{C3}
$$

where $\rho^2 = 1 + \tan^2\alpha + \tan^2\beta$ and the Jacobian of the transformation and the matrix $\tilde{\mathbf{A}}$ are, respectively,

$$\sqrt{g} = [\det(\mathbf{g}_{ij})]^{1/2} = \frac{R^2}{\rho^3 \cos^2\alpha \cos^2\beta}, \quad \text{and}$$

$$\tilde{\mathbf{A}} = \begin{bmatrix} R\cos\theta\lambda_\alpha & R\cos\theta\lambda_\beta \\ R\theta_\alpha & R\theta_\beta \end{bmatrix}. \tag{C4}$$

Note that for the equiangular projection, the metric tensor and other parameters defined above are independent of the constant a. The explicit form of $\tilde{\mathbf{A}}$ in (C4) and its inverse $\tilde{\mathbf{A}}^{-1}$ are needed for the transformation between cube faces and the sphere, and these are easily derived using the following relations and Eqs. (B10)–(B12):

$$\tilde{\mathbf{A}} = \mathbf{A}\begin{bmatrix} a/\cos^2\alpha & 0 \\ 0 & a/\cos^2\beta \end{bmatrix},$$

$$\tilde{\mathbf{A}}^{-1} = \frac{1}{a}\begin{bmatrix} \cos^2\alpha & 0 \\ 0 & \cos^2\beta \end{bmatrix}\mathbf{A}^{-1}. \tag{C5}$$

An alternative form of (C3) in terms of a, x, and y is

$$g_{ij} = \frac{a^2(1 + x^2)(1 + y^2)}{r^4}\begin{bmatrix} a^2 + x^2 & -xy \\ -xy & a^2 + y^2 \end{bmatrix}. \tag{C6}$$

REFERENCES

Biswas, R., K. Devine, and J. Flaherty, 1994: Parallel, advaptive finite element methods for conservation laws. *Appl. Numer. Maths.,* **14,** 255–283.

Cockburn, B., and C. W. Shu, 1989: TVB Runge–Kutta local projection discontinuous Galerkin method for conservation laws. II: General framework. *Math. Comput.,* **52,** 411–435.

——, and ——, 1998: The Runge–Kutta discontinuous Galerkin finite element method for conservation laws. V: Multidimensional systems. *J. Comput. Phys.,* **141,** 199–224.

——, and ——, 2001: The Runge–Kutta discontinuous Galerkin method for convection-dominated problems. *J. Sci. Comput.,* **16,** 173–261.

——, G. E. Karniadakis, and C. W. Shu, Eds., 2000: *Discontinuous Galerkin Methods: Theory, Computation, and Applications.* Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, 470 pp.

Doswell, C. A., III, 1984: A kinematic analysis of frontogenesis associated with a nondivergent vortex. *J. Atmos. Sci.,* **41,** 1242–1248.

Dutton, J. A., 1986: *Dynamics of Atmospheric Motion.* Dover, 617 pp.

Fournier, A., M. A. Taylor, and J. J. Tribbia, 2004: A spectral element atmospheric model (SEAM): High-resolution parallel computation and localized resolution of regional dynamics. *Mon. Wea. Rev.,* **132,** 726–748.

Giraldo, F. X., and T. E. Rosmond, 2004: A scalable spectral element Eulerian atmospheric model (SEE-AM) for NWP: Dynamical core test. *Mon. Wea. Rev.,* **132,** 133–153.

——, J. S. Hesthaven, and T. Warburton, 2002: Nodal high-order discontinuous Galerkin methods for spherical shallow water equations. *J. Comput. Phys.,* **181,** 499–525.

Gottlieb, S., C. W. Shu, and E. Tadmor, 2001: Strong stability-preserving high-order time discretization methods. *SIAM Rev.,* **43,** 89–112.

Henderson, R. D., 1999: Adaptive spectral element method for turbulence and transition. *High-Order Methods for Computational Physics*, T. J. Barth and H. Deconinck, Eds., Lecture Notes in Computational Science and Engineering, Vol. 9, Springer-Verlag, 225–324.

Karniadakis, G. E., and S. J. Sherwin, 1999: *Spectral/hp Element Methods for CFD.* Oxford University Press, 390 pp.

Lesaint, P., and P. A. Raviart, 1974: On finite-element methods for solving the neutron transport equations. *Mathematical Aspects of Finite Elements in Partial Differential Equations*, C. de Boor, Ed., Academic Press, 89–145.

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

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.

——, J. Côté, and A. Staniforth, 1999: Cascade interpolation for semi-Lagrangian advection over the sphere. *Quart. J. Roy. Meteor. Soc.,* **125,** 1445–1468.

——, J. S. Scroggs, and F. H. M. Semazzi, 2003: A forward trajectory semi-Lagrangian scheme over the sphere. *J. Comput. Phys.,* **190,** 275–294.

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

Randall, D. A., T. D. Ringler, R. P. Heikes, P. Jones, and J. Baumgardner, 2002: Climate modeling with spherical geodesic grids. *Comput. Sci. Eng.,* **4,** 32–41.

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.

Shu, C.-W., 1988: Total variation diminishing time discretizations. *SIAM J. Sci. Stat. Comput.,* **9,** 1073–1084.

Taylor, M., J. Tribbia, and M. Iskandrani, 1997: The spectral element method for the shallow water equations on the sphere. *J. Comput. Phys.,* **130,** 92–108.

Thomas, S. J., and R. D. Loft, 2002: Semi-implicit spectral element model. *J. Sci. Comput.,* **17,** 339–350.

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