INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2014; **75**:103–133
Published online 19 February 2014 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/fld.3887

A high-order fully explicit flux-form semi-Lagrangian shallow-water model

Paul Aaron Ullrich[1,*,†], Peter Hjort Lauritzen[2] and Christiane Jablonowski[3]

[1]*Department of Land, Air and Water Resources, University of California, Davis, One Shields Avenue, Davis, CA 95616, USA*
[2]*Climate and Global Dynamics Division, National Center for Atmospheric Research, Boulder, CO, USA*
[3]*Department of Atmospheric, Oceanic and Space Sciences, Space Research Building, University of Michigan, 2455 Hayward St., Ann Arbor, MI 48109, USA*

SUMMARY

A new approach is proposed for constructing a fully explicit third-order mass-conservative semi-Lagrangian scheme for simulating the shallow-water equations on an equiangular cubed-sphere grid. State variables are staggered with velocity components stored pointwise at nodal points and mass variables stored as element averages. In order to advance the state variables in time, we first apply an explicit multi-step time-stepping scheme to update the velocity components and then use a semi-Lagrangian advection scheme to update the height field and tracer variables. This procedure is chosen to ensure consistency between dry air mass and tracers, which is particularly important in many atmospheric chemistry applications. The resulting scheme is shown to be competitive with many existing numerical methods on a suite of standard test cases and demonstrates slightly improved performance over other high-order finite-volume models. Copyright © 2014 John Wiley & Sons, Ltd.

Received 27 February 2013; Revised 3 December 2013; Accepted 25 January 2014

KEY WORDS: high order; semi-lagrangian; finite-volume methods; shallow-water equations

1. INTRODUCTION

In the past decade, the speed of individual computer processors has flatlined, leading to new hardware architecture based around large-scale multi-processor supercomputing systems. This development has led to the need for new software that performs well on these systems. In particular, it has become increasingly important to design atmospheric models that are capable of scaling on systems with tens to hundreds of thousands of processors. Many projects that tackle this problem are currently underway, including the Community Atmosphere Model spectral element dynamical core [1, 2] based on the work of Taylor *et al*. [3] and the Geophysical Fluid Dynamics Laboratory cubed-sphere finite-volume model [4, 5] based on the work of Putman and Lin [6], among others. In this paper, we present one such effort in designing a novel numerical method that tackles this problem, here utilizing the conservative semi-Lagrangian multi-tracer scheme (CSLAM) of [7, 8] to solve the shallow-water equations on a cubed-sphere grid. Building a model atop the infrastructure of CSLAM immediately leads to an intrinsic consistency of the total mass and tracer mass fields, among other desirable properties (e.g., [9]).

*Correspondence to: Paul Aaron Ullrich, Department of Land, Air and Water Resources, University of California, Davis, One Shields Avenue, Davis, CA 95616, USA.
†E-mail: paullrich@ucdavis.edu

Modeling the shallow-water equations is an important first step in understanding how a discretization of the atmospheric primitive equations would behave in practice. The shallow-water equations are the simplest equation set to maintain the defining features of atmospheric motions, such as Rossby waves and inertia–gravity waves, without the added complexity of the vertical dimension. Many numerical schemes have previously been developed that model these equations on the sphere. Other models include the spectral transform method [10], finite-difference approaches [11–13], high-order finite-volume methods [14], hybrid finite-difference/finite-volume methods [15, 16], and finite-element models [3, 17–20]. Closely related to the current effort is the work in [21], which describes a semi-implicit shallow-water model in Cartesian geometry based on CSLAM.

In contrast with other semi-Lagrangian shallow-water models, the method described in this paper uses an inherently conservative flux-form formulation to guarantee conservation of mass. The velocity field is updated using a third-order multi-step time-stepping procedure (third-order extrapolated backward differentiation formulas (eBDF)) along with a flow-dependent fourth-order hyperdiffusion operator to maintain stability. In conjunction with the quasi-uniform cubed-sphere grid, this combination maintains locality of the update operation, at the cost of a strict CFL condition imposed by the fully explicit update. Time-stepping procedures that allow for a more lenient time-step constraint, such as semi-implicit methods [22] or Laplace transforms [23], were not explored in this work because they do not preserve strict locality.

The explicit time-stepping procedure described in this paper is applicable to any conservative semi-Lagrangian transport scheme, including the family of semi-Lagrangian integrated-mass schemes [24], the Semi-Lagrangian Inherently Conserving and Efficient scheme [25–27], and certain dimension-split semi-Lagrangian formulations [28, 29]. However, the flux-form implementation of CSLAM is particularly well suited for implementation as a dynamical core on the cubed-sphere grid because it does not use dimension splitting (which is potentially problematic near cubed-sphere corner points) and can be modified to guarantee formal third-order accuracy [30]. In particular, the use of quadratic upstream edges [30] is important for avoiding errors in the numerically computed divergence, which can pollute the solution.

Quasi-uniform meshes are of particular interest, as their quasi-uniformity significantly reduces the complexity of developing numerical methods capable of scaling to a large number of processors. The regular latitude–longitude (RLL) grid, for instance, does not share the quasi-uniformity property because it suffers from convergence of grid lines at the poles. Consequently, gridpoint-based numerical methods using the RLL grid require filtering to maintain numerical stability. The cubed-sphere grid, which is harnessed by the model in this paper, has been the basis for models that scale well on massively parallel computing platforms [5, 31]. Parallel scalability for models using the cubed-sphere grid is ensured as long as communication is sufficiently localized: that is, as long as the discretization only requires communication with a small number of processing units, independent of the total number of processors used for computing the simulation.

The paper is organized as follows. In Section 2, we introduce the cubed-sphere grid under the equiangular projection, followed by a discussion of the shallow-water equations in cubed-sphere geometry in section 3. Evolution of the velocity field is described in Section 4. The discretization of the advected components of the flow, which includes the continuity equation and any tracer fields, is discussed in Section 5. Numerical results from this method are presented in Section 6, followed by our conclusions in Section 7.

2. THE CUBED SPHERE

The cubed-sphere grid [12, 32] consists of a cube with six Cartesian patches arranged along each face, which is then 'inflated' to fill a spherical shell. On the equiangular cubed-sphere grid, coordinates are given as (α, β, n_p), with central angles $\alpha, \beta \in [-\pi/4, \pi/4]$ and panel index $n_p \in \{1, 2, 3, 4, 5, 6\}$. By convention, we choose panels 1–4 to be along the equator and panels 5 and 6 to be centered on the northern and southern poles, respectively. Gnomonic coordinates are related to equiangular coordinates via the transform

$$X = \tan\alpha, \qquad Y = \tan\beta. \tag{1}$$

DOI: 10.1002/fld

Table I. Properties of the cubed-sphere grid for different resolutions.

Resolution	Δx (km)	A_{avg} (km^2)	$A_{\text{min}}/A_{\text{max}}$	RLL_{equiv} (°)	T_{equiv}
c20	500.0	2.125×10^5 km^2	0.7359	5.2°	$T21$
c40	250.0	5.313×10^4 km^2	0.7213	2.6°	$T42$
c80	125.0	1.328×10^4 km^2	0.7141	1.3°	$T85$
c160	62.5	3.321×10^3 km^2	0.7106	0.65°	$T170$

Here, Δx is the grid spacing at the equator, A_{avg} is the average area of all cubed-sphere grid elements, A_{min} is the minimum element area, and A_{max} is the maximum element area. RLL_{equiv} denotes the equivalent grid spacing (in degrees) on the regular latitude–longitude grid with the same number of elements, and T_{equiv} denotes the approximate triangular truncation of a spectral transform method.

Gnomonic coordinates are particularly useful because any straight line in gnomonic coordinates is also a great circle arc, which is not the case for general line segments in equiangular coordinates. Further, integration over regions in gnomonic coordinates is typically much simpler than in equiangular coordinates. Throughout this paper, we will be making use of the metric term

$$\delta = \left[1 + \tan^2 \alpha + \tan^2 \beta\right]^{1/2}, \tag{2}$$

which appears frequently in geometric calculations on the cubed-sphere grid.

The discrete resolution of the cubed sphere is typically written in the form $c\langle N_c \rangle$, where each coordinate direction consists of N_c grid elements. Hence, the total number of grid elements on the cubed sphere is $N_c \times N_c \times 6$. Grid elements on a particular panel are denoted by $\mathcal{Z}_{i,j}$ with indices $(i, j) \in [0, \ldots, N_c - 1]^2$, which refers to the region bounded by

$$\alpha \in \left[i\,\Delta\alpha - \frac{\pi}{4}, (i+1)\Delta\alpha - \frac{\pi}{4}\right], \qquad \beta \in \left[j\Delta\beta - \frac{\pi}{4}, (j+1)\Delta\beta - \frac{\pi}{4}\right], \tag{3}$$

where on an equiangular grid, the grid spacing is

$$\Delta\alpha = \Delta\beta = \pi/(2N_c). \tag{4}$$

Equiangular element center points are defined for each element as the point $\boldsymbol{\alpha}_{i,j} = (\alpha_i, \beta_j)$ with

$$\alpha_i = \left(i + \frac{1}{2}\right)\Delta\alpha - \frac{\pi}{4}, \qquad \beta_j = \left(j + \frac{1}{2}\right)\Delta\beta - \frac{\pi}{4}. \tag{5}$$

Some properties of the cubed-sphere grid for a variety of resolutions are given in Table I. For a comprehensive mathematical description of the equiangular cubed-sphere grid, see [20, Appendices A–C] or [33, Appendices A and B].

3. THE SHALLOW-WATER EQUATIONS ON THE CUBED SPHERE

The shallow-water equations on the sphere can be concisely formulated using two equations. The first, known as the *continuity equation*, effectively describes the principle of conservation of mass as it moves under some background flow field. In a Eulerian or fixed frame, the conservative form of this equation takes the form

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{u}) = 0, \tag{6}$$

DOI: 10.1002/fld

where h denotes the fluid depth and $\mathbf{u}$ denotes the underlying horizontal velocity vector. Global conservation of h is immediately evident upon integrating this equation over the surface of the sphere and applying Gauss' divergence theorem, leading to

$$\frac{d}{dt}\int h dA = 0, \tag{7}$$

where dA is an appropriately chosen area element. The second shallow-water equation describes the evolution of the velocity field and can be expressed as

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = \boldsymbol{S}_C + \boldsymbol{S}_P, \tag{8}$$

where $\boldsymbol{S}_C$ and $\boldsymbol{S}_P$ respectively denote forcing terms due to the Coriolis force and variations in the fluid pressure (due to the height field and underlying topography). Using the standard unit-length radial coordinate vector $\mathbf{g}_r$, $\boldsymbol{S}_C$ can be written as

$$\boldsymbol{S}_C = -f\,\mathbf{g}_r \times \mathbf{u}, \tag{9}$$

where $f = 2\Omega\sin\varphi$ is the Coriolis parameter in terms of the rotation rate $\Omega = 7.292\times 10^{-5}$ s^{-1} and latitude φ. Similarly, $\boldsymbol{S}_P$ can be written as

$$\boldsymbol{S}_P = -g\nabla H, \tag{10}$$

where $g = 9.80616$ m s^{-2} represents the mean constant of gravity at the Earth's surface and $H = h + z$ denotes the free-surface height, which is the sum of the fluid depth h and the height of the bottom topography z.

Under equiangular coordinates, the velocity field is written as

$$\mathbf{u} = u^\alpha \mathbf{g}_\alpha + u^\beta \mathbf{g}_\beta, \tag{11}$$

where $\mathbf{g}_\alpha = (\partial\mathbf{x}/\partial\alpha)_\beta$ and $\mathbf{g}_\beta = (\partial\mathbf{x}/\partial\beta)_\alpha$ have units of length and denote the natural basis vectors of the underlying coordinate system (under Cartesian coordinates $\mathbf{x}(\alpha,\beta) = (x(\alpha,\beta), y(\alpha,\beta), z(\alpha,\beta))$). The coefficients u^α and u^β are known as the contravariant components of the velocity vector and have units of radian per second in the natural basis. This choice allows us to write the velocity evolution equation (8) in component form

$$\frac{\partial u^\alpha}{\partial t} + \mathbf{u}\cdot\nabla u^\alpha = S_C^\alpha + S_P^\alpha, \tag{12}$$

$$\frac{\partial u^\beta}{\partial t} + \mathbf{u}\cdot\nabla u^\beta = S_C^\beta + S_P^\beta. \tag{13}$$

By writing the covariant derivative as $\nabla_i u^j = (\partial u^j/\partial x^i) + \Gamma^j_{ik}u^k$ (where Γ^j_{ik} are the Christoffel symbols of the second kind in cubed-sphere coordinates and summation is implied over the repeated index k), we can express the advective term as

$$\mathbf{u}\cdot\nabla u^\alpha = u^\alpha\frac{\partial u^\alpha}{\partial\alpha} + u^\beta\frac{\partial u^\alpha}{\partial\beta} + \frac{2XY^2}{\delta^2}(u^\alpha)^2 - \frac{2Y\left(1+Y^2\right)}{\delta^2}u^\alpha u^\beta, \tag{14}$$

$$\mathbf{u}\cdot\nabla u^\beta = u^\alpha\frac{\partial u^\beta}{\partial\alpha} + u^\beta\frac{\partial u^\beta}{\partial\beta} - \frac{2X\left(1+X^2\right)}{\delta^2}u^\alpha u^\beta + \frac{2X^2Y}{\delta^2}\left(u^\beta\right)^2. \tag{15}$$

DOI: 10.1002/fld

In component form, the Coriolis term takes the form

$$\boldsymbol{S}_C = \begin{pmatrix} S_C^\alpha \\ S_C^\beta \end{pmatrix} = \frac{f}{\delta} \begin{pmatrix} -XYu^\alpha + \left(1+Y^2\right)u^\beta \\ -\left(1+X^2\right)u^\alpha + XYu^\beta \end{pmatrix}, \tag{16}$$

where the Coriolis parameter f can be written as

$$f = \begin{cases} \frac{2\Omega Y}{\delta} & \text{on equatorial panels } (n_p < 4), \\ \frac{2\Omega\sigma}{\delta} & \text{on polar panels.} \end{cases} \tag{17}$$

Here, σ is a panel indicator variable,

$$\sigma = \operatorname{sign}(\varphi) = \begin{cases} 1 & \text{on the northern panel } \left(n_p = 5\right), \\ -1 & \text{on the southern panel } \left(n_p = 6\right). \end{cases} \tag{18}$$

Finally, the forcing term due to fluid pressure is written in component form as

$$\boldsymbol{S}_P = \begin{pmatrix} S_P^\alpha \\ S_P^\beta \end{pmatrix} = -\frac{g\delta^2}{a^2\left(1+X^2\right)\left(1+Y^2\right)} \begin{pmatrix} \left(1+Y^2\right)\frac{\partial H}{\partial \alpha} + XY\frac{\partial H}{\partial \beta} \\ XY\frac{\partial H}{\partial \alpha} + \left(1+X^2\right)\frac{\partial H}{\partial \beta} \end{pmatrix}, \tag{19}$$

where a is the radius of the Earth ($a = 6.37122 \times 10^6$ m). Additional details about the derivation of the equations of motion in cubed-sphere coordinates can be found in [20].

4. EVOLUTION OF THE VELOCITY FIELD

The semi-Lagrangian shallow-water model makes use of the Arakawa B-grid [34], so that prognostic velocity points are located at the corners of each finite-volume element (nodes) and conserved variables (including air mass and tracer mass) are stored as element-averaged densities (Figure 1). Note that we store the element-averaged free-surface height H instead of the fluid depth h, as H is generally a smooth function, whereas rapid variations in the underlying topography may lead to rapid variations in h. This choice of grid also allows us to maximize the accuracy of the trajectory calculation over small time steps, which are unlikely to deviate far from element nodes. Although it is known that finite-difference methods suffer from poor dispersion properties on the B-grid as a consequence of its treatment of the divergent modes [35], the semi-Lagrangian approach handles divergence via area deformation and so is not subject to the same analysis. Further, the largest wave numbers are damped as a consequence of the incremental-remapping procedure and so should not be responsible for added noise.

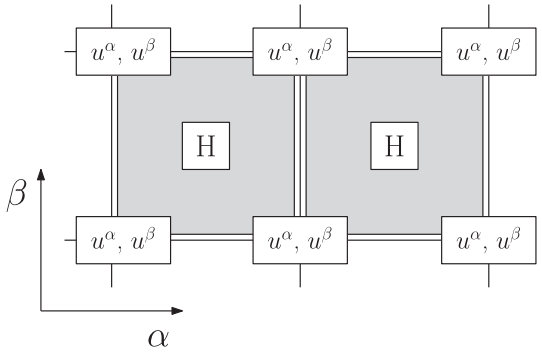

Figure 1. A depiction of the Arakawa B-grid used in the semi-Lagrangian shallow-water model. The free-surface height field and tracer mass are stored as element averages within each element, whereas the velocity field is stored pointwise at element nodes.

DOI: 10.1002/fld

Evolution of the velocity field is performed via time splitting, wherein the space and time components of the update equation (8) are discretized separately. Only the mass variables are exactly conserved under this approach; no explicit effort is made to conserve other invariant quantities, such as energy or potential vorticity.

4.1. Spatial discretization

To proceed, we first define a spatial operator $\boldsymbol{\mathcal{S}} = (\mathcal{S}^\alpha, \mathcal{S}^\beta)$, which allows us to write (12) and (13) as

$$\frac{\partial u^\alpha}{\partial t} = \mathcal{S}^\alpha \left(H, u^\alpha, u^\beta \right), \qquad \frac{\partial u^\beta}{\partial t} = \mathcal{S}^\beta \left(H, u^\alpha, u^\beta \right). \tag{20}$$

That is, $\boldsymbol{\mathcal{S}}$ incorporates advective terms, Coriolis forces, topographical forces, height (pressure) gradient forces, and artificial diffusion into a single operator that is applied to the height and velocity fields at a fixed point in time. We expand the spatial operator in terms of its individual contributions as

$$\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}}_A + \boldsymbol{\mathcal{S}}_C + \boldsymbol{\mathcal{S}}_P + \boldsymbol{\mathcal{S}}_D, \tag{21}$$

where the subscripts A, C, P, and D denote discretizations of advection (14–15), the Coriolis force (16), forcing due to fluid pressure (19), and artificial diffusion, respectively. Fourth-order central discretizations are used when derivatives are required within each of these expressions. In particular, the advection term (14 and 15) relies on derivatives of the velocity field along lines of constant α and β, which are computed as

$$\left(\frac{\partial u^c}{\partial \alpha} \right)_{i+1/2,j+1/2} \approx \frac{-u^c_{i+5/2,j+1/2} + 8u^c_{i+3/2,j+1/2} - 8u^c_{i-1/2,j+1/2} + u^c_{i-3/2,j+1/2}}{12\Delta\alpha}, \tag{22}$$

$$\left(\frac{\partial u^c}{\partial \beta} \right)_{i+1/2,j+1/2} \approx \frac{-u^c_{i+1/2,j+5/2} + 8u^c_{i+1/2,j+3/2} - 8u^c_{i+1/2,j-1/2} + u^c_{i+1/2,j-3/2}}{12\Delta\beta}, \tag{23}$$

where the superscript c represents either α or β. The fluid pressure term (19) requires fourth-order approximations to derivatives of the total height field at element nodes (denoted with half indices). These derivatives are computed via the 12-point stencil

$$\begin{aligned} \left(\frac{\partial H}{\partial \alpha} \right)_{i+1/2,j+1/2} \approx \frac{1}{24\Delta\alpha} \{ & \left[-\overline{H}_{i+2,j+1} + \overline{H}_{i-1,j+1} - \overline{H}_{i+2,j} + \overline{H}_{i-1,j} \right] \\ & + 2 \left[-\overline{H}_{i+1,j+2} + \overline{H}_{i,j+2} - \overline{H}_{i+1,j-1} + \overline{H}_{i,j-1} \right] \\ & + 17 \left[\overline{H}_{i+1,j+1} + \overline{H}_{i+1,j} - \overline{H}_{i,j+1} - \overline{H}_{i,j} \right] \}, \end{aligned} \tag{24}$$

$$\begin{aligned} \left(\frac{\partial H}{\partial \beta} \right)_{i+1/2,j+1/2} \approx \frac{1}{24\Delta\beta} \{ & \left[-\overline{H}_{i+1,j+2} + \overline{H}_{i+1,j-1} - \overline{H}_{i,j+2} + \overline{H}_{i,j-1} \right] \\ & + 2 \left[-\overline{H}_{i+2,j+1} + \overline{H}_{i+2,j} - \overline{H}_{i-1,j+1} + \overline{H}_{i-1,j} \right] \\ & + 17 \left[\overline{H}_{i+1,j+1} - \overline{H}_{i+1,j} + \overline{H}_{i,j+1} - \overline{H}_{i,j} \right] \}. \end{aligned} \tag{25}$$

Because the expression for the Coriolis force (16) contains no derivatives, it is simply evaluated pointwise at the element node.

DOI: 10.1002/fld

4.2. Artificial diffusion

The artificial diffusion term $\boldsymbol{\mathcal{S}}_D$ is required in our numerical discretization to preserve stability and prevent the formation of $2\Delta\alpha$ noise in the velocity field that may lead to a failure in the trajectory algorithm. The choice of artificial diffusion is not necessarily grounded in the physics of the problem but should instead depend on the properties of the numerical scheme and should target regions where errors in the advection and velocity evolution algorithms are most prominent. Motivated by these restrictions, we have settled on a choice of the artificial diffusion that takes the form

$$\boldsymbol{\mathcal{S}}_D = \frac{C_D \delta^4}{a^4} \begin{pmatrix} D_\alpha^4 u^\alpha + D_\beta^4 u^\alpha \\ D_\alpha^4 u^\beta + D_\beta^4 u^\beta \end{pmatrix}, \tag{26}$$

where C_D is the hyperdiffusion coefficient (with units m^4/s) and the operators D_α^4 and D_β^4 are approximations to the fourth α and β derivatives, respectively. This choice of diffusion operator is constructed to loosely approximate the fourth-order Laplacian for the velocity field on the cubed-sphere grid and is third-order accurate as long as C_D is at least proportional to $\Delta\alpha^3$.

Analogous to the differences of the velocity field used in the advection term, the fourth-order derivative operators are approximated as

$$\left(D_\alpha^4 u^c\right)_{i+1/2,j+1/2} \approx \frac{u^c_{i+5/2,j+1/2} - 4u^c_{i+3/2,j+1/2} + 6u^c_{i+1/2,j+1/2} - 4u^c_{i-1/2,j+1/2} + u^c_{i-3/2,j+1/2}}{\Delta\alpha^4}, \tag{27}$$

$$\left(D_\beta^4 u^c\right)_{i+1/2,j+1/2} \approx \frac{u^c_{i+1/2,j+5/2} - 4u^c_{i+1/2,j+3/2} + 6u^c_{i+1/2,j+1/2} - 4u^c_{i+1/2,j-1/2} + u^c_{i+1/2,j-3/2}}{\Delta\beta^4}. \tag{28}$$

The flow-dependent diffusion coefficient C_D is chosen to approximate numerical diffusion from the Rusanov flux function [36], leading to

$$C_D = -C_R \left(a|\mathbf{u}|_{i+1/2,j+1/2} + \sqrt{g h_{i+1/2,j+1/2}} \right), \tag{29}$$

where C_R has units of cubic meter and depends purely on the resolution of the grid, $|\mathbf{u}|$ denotes the magnitude of the velocity (in rad/s), and $h_{i+1/2,j+1/2}$ is a fourth-order approximation to the height field at this element node, given by

$$\begin{aligned} h_{i+1/2,j+1/2} = \frac{1}{24} \{ & 8 \left[\overline{H}_{i+1,j+1} + \overline{H}_{i,j+1} + \overline{H}_{i+1,j} + \overline{H}_{i,j} \right] \\ & - \left[\overline{H}_{i+1,j+2} + \overline{H}_{i,j+2} + \overline{H}_{i+2,j+1} + \overline{H}_{i-1,j+1} \right. \\ & \left. + \ \overline{H}_{i+2,j} + \overline{H}_{i-1,j} + \overline{H}_{i+1,j-1} + \overline{H}_{i,j-1} \right] \} - z_{i+1/2,j+1/2}. \end{aligned} \tag{30}$$

Observe that the term within the brackets in (29) is simply the maximum shallow-water wave speed, which corresponds to the maximum eigenvalue of the exact spatial operator. The values of C_R used in this paper are given in Table II for several cubed-sphere grid resolutions. These values are determined empirically by selecting the smallest diffusion coefficient that leads to a stable numerical scheme. For a field with $h(\alpha, \beta) = 8000$ m, these coefficients are approximately five times smaller than the default values used in the Community Atmosphere Model spectral element dynamical core.

DOI: 10.1002/fld

Table II. Value of the coefficient C_R used at several possible cubed-sphere grid resolutions.

Resolution	C_R (m^3)
c20	2.0×10^{14}
c40	1.3×10^{13}
c80	1.1×10^{12}
c160	1.3×10^{11}

4.3. Temporal discretization

Time evolution of the velocity field is performed using a high-order multi-step method in order to preserve locality of the update operation. A multi-step method is needed for the velocity update, as multi-stage methods such as the Runge–Kutta family of schemes require information on the h field over fractional time steps, which is not available until the velocity update has completed. The time update makes use of the spatial operator $\boldsymbol{\mathcal{S}}$ evaluated at time level t^n, which we denote

$$\boldsymbol{\mathcal{S}}^n = \boldsymbol{\mathcal{S}}\left(H^n, (u^\alpha)^n, (u^\beta)^n\right). \tag{31}$$

The eBDF [37] are preferred in our case, as they provide a good balance between stability and accuracy. The third-order eBDF scheme takes the form

$$\mathbf{u}^{n+1} = \frac{18}{11}\mathbf{u}^n - \frac{9}{11}\mathbf{u}^{n-1} + \frac{2}{11}\mathbf{u}^{n-2} + \frac{18}{11}\Delta t\boldsymbol{\mathcal{S}}^n - \frac{18}{11}\Delta t\boldsymbol{\mathcal{S}}^{n-1} + \frac{6}{11}\Delta t\boldsymbol{\mathcal{S}}^{n-2}. \tag{32}$$

Because multi-step methods require an initial start-up, we make use of the forward Euler method initially,

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t\boldsymbol{\mathcal{S}}^n, \tag{33}$$

and at the second time step use the second-order Adams Bashforth scheme (AB2),

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \frac{3}{2}\Delta t\boldsymbol{\mathcal{S}}^n - \frac{1}{2}\Delta t\boldsymbol{\mathcal{S}}^{n-1}. \tag{34}$$

For reasons of efficiency, the velocity field $\mathbf{u}$ and the spatial operator $\boldsymbol{\mathcal{S}}^n$ are stored at time levels t^{n-1} and t^{n-2} and then reused in subsequent update operations.

5. CONSERVATIVE SEMI-LAGRANGIAN ADVECTION

Once the velocity field has been updated, all conserved mass fields are then advanced via passive advection. Using the updated velocity field, we compute trajectories backwards in time so as to obtain a *flux volume* associated with each edge. Integrating the total tracer mass within each flux volume then yields the total mass that has passed through that edge over the duration of the time step. These mass fluxes are then used to update the element averages.

5.1. Treatment of advection

The advection scheme we use for this work is the Gaussian quadrature variant of the simplified flux-form CSLAM scheme [30]. The advection component is used for updating the height field plus any tracer fields that are transported by the velocity field.

The conservative semi-Lagrangian method we use in this paper is based on the simplified flux-form semi-Lagrangian formulation [38], which has been recently implemented on the cubed sphere [39]. Under this approach, we begin by considering the conservation law

DOI: 10.1002/fld

$$\frac{\partial \psi}{\partial t} + \nabla \cdot (\psi \mathbf{u}) = 0, \tag{35}$$

where $\psi = h$ for the shallow-water height equation (6) or $\psi = h\phi$ for an arbitrary tracer with mixing ratio ϕ. We integrate the conservation law over an arbitrary grid cell $\mathcal{Z}_{i,j}$ and apply Gauss' divergence theorem to obtain

$$\frac{d\overline{\psi}_{i,j}}{dt} = -\frac{1}{|\mathcal{Z}_{i,j}|} \oint_{\partial \mathcal{Z}_{i,j}} \mathbf{F} \cdot d\mathbf{S}, \tag{36}$$

where $\partial \mathcal{Z}_{i,j}$ denotes the boundary of $\mathcal{Z}_{i,j}$, $|\mathcal{Z}_{i,j}|$ is the area of $\mathcal{Z}_{i,j}$, and $\mathbf{F} \cdot d\mathbf{S}$ denotes the pointwise outward flux through the boundary. The overbar denotes an area average of the form

$$\overline{\psi} = \frac{1}{|\mathcal{Z}_{i,j}|} \int_{\mathcal{Z}_{i,j}} \psi dV, \tag{37}$$

where $dV = Jd\alpha d\beta$ is the volume element in equiangular coordinates, with

$$J = \frac{(1 + \tan^2 \alpha)(1 + \tan^2 \beta)}{\delta^3}. \tag{38}$$

Integrating (36) in time from t^n to t^{n+1} gives

$$\overline{\psi}_{i,j}^{n+1} = \overline{\psi}_{i,j}^{n} - \frac{1}{|\mathcal{Z}_{i,j}|} \int_{t^n}^{t^{n+1}} \oint_{\partial \mathcal{Z}_{i,j}} \mathbf{F} \cdot d\mathbf{S} dt. \tag{39}$$

Because $\mathcal{Z}_{i,j}$ is a rectangular region in computational (α, β) space, we can denote its four edges by the compass directions east (E), north (N), west (W), and south (S). Hence, (39) can be written in the form

$$\overline{\psi}_{i,j}^{n+1} = \overline{\psi}_{i,j}^{n} + \frac{1}{|\mathcal{Z}_{i,j}|} \left[F_E + F_N + F_W + F_S \right], \tag{40}$$

where F_E is the total mass flux *into the element* through the east edge from time t^n to t^{n+1}, and similarly for the north, west, and south edges. Note that the fluxes can be positive or negative, depending on the direction of advection. Under the flux-form semi-Lagrangian formulation (Figure 2), these fluxes are computed by integrating over the flux volume for each edge. This flux volume consists of a region of the simulation domain bounded on one side by the corresponding fixed edge of $\mathcal{Z}_{i,j}$ at time t^{n+1} and on the other by the numerically computed upstream projection of that edge at time t^n. The four regions associated with the east, north, west, and south edges are depicted in Figure 2c–f, respectively, and denoted by $\mathbf{a}_{i,j}^{\tau=1}$ through $\mathbf{a}_{i,j}^{\tau=4}$. For example, for the east edge, we have

$$F_E = \sigma_E \int_{\mathbf{a}_k^{\tau=1}} \psi^n(\mathbf{X}) dV, \tag{41}$$

where σ_E indicates the direction of the flux (positive or negative depending on whether flow is into or out of the element), $\mathbf{a}_k^{\tau=1}$ is the upstream flux volume defined by Figure 2c, and dV is a volume element. Summation over sub-volumes is implied if the flux volume consists of both inward and outward components.

The accuracy of the semi-Lagrangian advection scheme partially relies on obtaining an accurate approximation of the flux volume. The use of remapping-based methods for transport is known to introduce spurious divergence into an otherwise divergence-free flow [40]. These errors are especially evident for divergence-free sheared flows on the cubed sphere, which pass near panel corner points. In this region, small defects in the divergence can easily drive wavenumber 4 imprinting from

DOI: 10.1002/fld

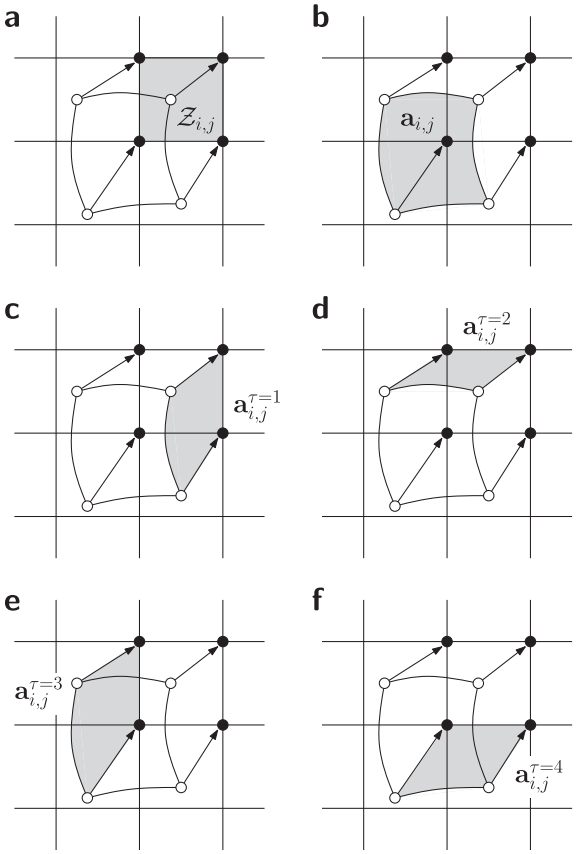

Figure 2. An illustration of the semi-Lagrangian approach. The nodes of element $\mathcal{Z}_{i,j}$ (a) are tracked upstream to element $\mathbf{a}_{i,j}$ (b). Under the flux-form formulation, we integrate over flux areas for the (c) east $\mathbf{a}_{i,j}^{\tau=1}$, (d) north $\mathbf{a}_{i,j}^{\tau=2}$, (e) west $\mathbf{a}_{i,j}^{\tau=3}$, and (f) south $\mathbf{a}_{i,j}^{\tau=4}$ face. The original element (a) plus the sum of all flux area masses (c–f), weighted by the direction of the flow, are equal to the mass in the upstream element $\mathbf{a}_{i,j}$ regardless of the detailed shape of the trajectory.

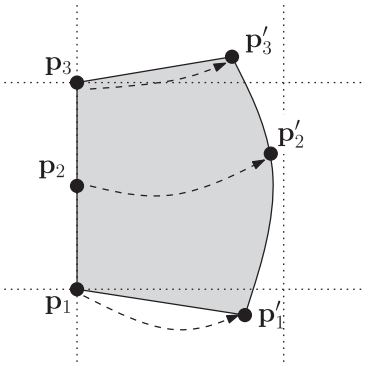

Figure 3. An illustration of a third-order approximation to an upstream flux volume, which has been obtained by tracking the edge $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ upstream to points $(\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3)$. A quadratic curve is fit through $(\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3)$ so as to approximate the upstream projection of the edge. The dashed arrows denote backwards trajectories, in this case corresponding to a westward flow.

the underlying grid and reduce the formal accuracy of the transport scheme to the second order [30]. In order to avoid these issues, we follow [30] and instead approximate the upstream source region with a quadratic curve, which dramatically reduces the appearance of spurious divergence and

DOI: 10.1002/fld

further guarantees third-order convergence. These curves can be reconstructed by interpolating a quadratic through three nodal points, which include the upstream projection of the two endpoints of each edge plus the upstream projection of the edge centerpoint (Figure 3). Each of the nodal points depicted in this Figure 3 can be written in terms of gnomonic coordinates as $\mathbf{p}_i = (X_i, Y_i)$ for nodes along the flux edge and $\mathbf{p}'_i = (X'_i, Y'_i)$ for nodes along the upstream edge.

Further, the flux volume is not necessarily convex and, in fact, can be quite deformed depending on the trajectory of each edge. All possibilities that are allowed by our model have been enumerated in Figure 4 for an edge that bounds the eastern side of an element. In this figure, regions that lead to an outward mass flux ($\sigma_E = -1$) are lightly shaded, whereas regions that lead to an inward mass flux ($\sigma_E = 1$) are heavily shaded.

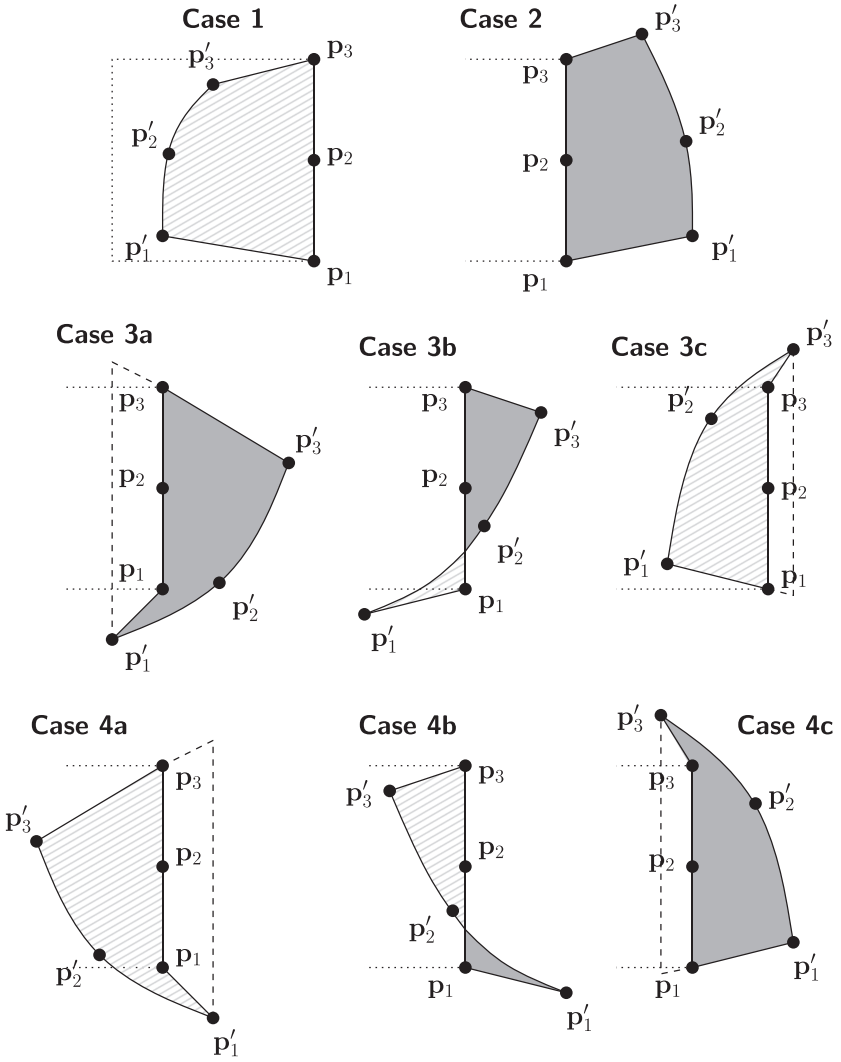

Figure 4. Depiction of each of the cases that must be treated by the flux integration algorithm, here with an east edge as example. The edge $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ denotes the edge for which the advective flux is desired. The points $(\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3)$ are determined by the upstream trajectory computation algorithm from points $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$. All edges are straight-line segments except for $(\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3)$, which is a quadratic curve. Regions that lead to a positive flux across the edge are lightly shaded, whereas regions that lead to a negative flux across the edge are heavily shaded. The dotted lines denote the north/south edges of the left element.

DOI: 10.1002/fld

5.2. Integration of the flux volume

In this section, we discuss the numerical procedure for evaluating the integral of the flux volume (41). We assume that a gnomonic sub-grid-scale reconstruction of the form

$$\psi_{i,j}(X,Y) = \sum_{p,q} c_{(p,q)} X^p Y^q \tag{42}$$

is known within each element $\mathcal{Z}_{i,j}$, such as the one described in Appendix A. Here, $c_{(p,q)}$ are the reconstruction coefficients associated with the sub-grid-scale reconstruction, and the range of the indices (p,q) is determined by the choice of reconstruction. Following the simplified flux-form approach [30, 38, 39], integration of the flux volume is only performed over the sub-grid-scale reconstruction of neighboring elements. For example, if a given edge separates elements $\mathcal{Z}_{i,j}$ and $\mathcal{Z}_{i+1,j}$, then in accordance with Figure 4, the sub-grid-scale reconstruction $\psi_{i,j}(X,Y)$ is used for computing the mass in the lightly shaded regions, and $\psi_{i+1,j}(X,Y)$ is used for computing the mass in the heavily shaded regions. This approach greatly simplifies the integration procedure because there is no need to identify where overlaps between the fixed grid and flux volume occur. Note that if the given edge is also a cubed-sphere panel edge, the integration must be applied on the correct panel to ensure consistency.

Several options exist for computing the integrals over the flux volume. Under the quadrature-based formulation [30], these integrals are computed via an appropriately chosen quadrature rule. For fourth-order accuracy, a four-point quadrature rule, such as the one described later, can be used. In cases 3 and 4 of Figure 4, the convex property of the generalized quadrilaterals has been lost, and so a single quadrature rule for each flux volume is insufficient. Instead, for cases 3b and 4b, we break the flux volume into two sub-volumes and, treating these as generalized quadrilaterals, simply apply the quadrature rule to each sub-volume. For cases 3a, 3c, 4a, and 4c, the integral is instead over a wedge-shaped region. To integrate over the wedge, we extend the integration region outward to form a generalized triangular region (dashed lines) so that the wedge can be viewed as the difference between the generalized triangle and a convex quadrilateral region. A quadrature rule is then applied to both regions and the difference taken to obtain the integral over the wedge. In using this approach for dividing the flux region, two corner points of our integration domain will always lie along a line of constant X. A four-point quadrature rule is used in this work to ensure fourth-order integration of the upstream flux area, which is described in Appendix B.

5.3. Computing trajectories

To track the nodal points backwards in time (Figure 3), we have made use of the definition of the velocity field in Lagrangian form. That is,

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x}, t), \tag{43}$$

where $\mathbf{x} = (\alpha, \beta)$ is a coordinate vector and $\mathbf{u} = (u^\alpha, u^\beta)$ is the vector of contravariant wind vector components. Because this equation is simply a coupled system of two ordinary differential equations, we can easily discretize it using, for instance, the fourth-order Runge–Kutta scheme

$$\mathbf{x}^{(1)} = \mathbf{x}_0 - \frac{\Delta t}{2}\mathbf{u}\left(\mathbf{x}_0, t^{n+1}\right), \tag{44}$$

$$\mathbf{x}^{(2)} = \mathbf{x}_0 - \frac{\Delta t}{2}\mathbf{u}\left(\mathbf{x}^{(1)}, t^{n+1} - \frac{\Delta t}{2}\right), \tag{45}$$

$$\mathbf{x}^{(3)} = \mathbf{x}_0 - \Delta t\,\mathbf{u}\left(\mathbf{x}^{(2)}, t^{n+1} - \frac{\Delta t}{2}\right), \tag{46}$$

$$\mathbf{x}^{(4)} = -\frac{1}{3}\mathbf{x}_0 + \frac{1}{3}\mathbf{x}^{(1)} + \frac{2}{3}\mathbf{x}^{(2)} + \frac{1}{3}\mathbf{x}^{(3)} - \frac{\Delta t}{6}\mathbf{u}\left(\mathbf{x}^{(3)}, t^n\right). \tag{47}$$

DOI: 10.1002/fld

The coordinates of $\mathbf{x}^{(4)}$ are then adopted as the 'origin' of the point $\mathbf{x}_0$. In particular, $\mathbf{x}^{(4)}$ is a fourth-order accurate approximation to the point at time t^n that, on undergoing Lagrangian advection, would have ended up at $\mathbf{x}_0$ at time t^{n+1}. Notably, this approach only requires knowledge of the velocity field at time t^{n+1}, $t^{n+1} - \Delta t/2$ and $t^n = t^{n+1} - \Delta t$. Because the velocity field is already known at time t^{n+1} and t^n from the velocity evolution procedure, it only remains for us to obtain an approximation to the velocity field at the half time step $t^{n+1} - \Delta t/2$. Many possible polynomial interpolants may be defined through all stored velocity fields, but we have found that a simple average of the velocity field at t^n and t^{n+1} is sufficient for accuracy and stability of this method. That is, we make the approximation

$$\mathbf{u}\left(\mathbf{x}, t^{n+1} - \frac{\Delta t}{2}\right) = \frac{\mathbf{u}(\mathbf{x}, t^{n+1}) + \mathbf{u}(\mathbf{x}, t^n)}{2}. \tag{48}$$

5.4. Monotonicity and positivity

For the advective scheme, positivity-preserving and monotonicity-preserving limiters have been implemented and are described in Appendix A.2. The limiting algorithm is identical to that used by [7] and [41]. These limiters are relatively inexpensive and do not require additional parallel communications. However, for a flux-form semi-Lagrangian method, this approach only guarantees positivity or monotonicity up to a Courant number of 0.5. An alternative approach that is compatible with the flux-form formulation of this scheme would use flux-corrected transport (FCT) [8, 42]. The use of FCT guarantees unconditional monotonicity or positivity but typically requires one additional parallel communication per time step.

5.5. Stability considerations

The stability of the scheme described in this paper has been verified empirically using energy estimates and long-duration simulations. For the shallow-water variables $(h, \mathbf{u})$, the maximum stable time-step size is limited by the critical wave speed $u_c = |\mathbf{u}| + \sqrt{gh}$. Instability generally manifests via the appearance of $2\Delta x$ noise in the velocity field, which can lead to crossed trajectories and subsequent breakdown of the simulation. Repeated tests suggest a maximum CFL limit of approximately 0.35, which is competitive with the maximum per-stage CFL limit for a typical high-order finite-volume scheme with a third-order three-stage Runge–Kutta time discretization. The CFL condition for pure flux-form semi-Lagrangian advection (with prescribed velocities) is approximately 0.75 [39].

6. NUMERICAL RESULTS

In this section, we present numerical results that demonstrate accuracy, stability, and consistency of the numerical method for several standard shallow-water test problems. These problems are primarily from the well-known test suite in [43], although the barotropic instability test presented in Section 6.6 is from [44].

When required, standard error measures are calculated via

$$L_1(h) = \frac{I\left[|h - h_T|\right]}{I\left[|h_T|\right]}, \tag{49}$$

$$L_2(h) = \sqrt{\frac{I\left[(h - h_T)^2\right]}{I\left[h_T^2\right]}}, \tag{50}$$

$$L_\infty(h) = \frac{\max|h - h_T|}{\max|h_T|}, \tag{51}$$

DOI: 10.1002/fld

where h_T is the height field at the initial time (which is the analytical solution for steady-state test cases) and I denotes an approximation to the global integral, given by

$$I[x] = \sum_{\text{all cells } k} x_k A_k, \tag{52}$$

with A_k denoting the area of element k. For advection of a cosine bell (Section 6.1), we also make use of the relative minimum and maximum,

$$\langle \text{Relative minimum} \rangle = \frac{\min h - \min h_T}{\max |h_T|}, \tag{53}$$

$$\langle \text{Relative maximum} \rangle = \frac{\max h - \max h_T}{\max |h_T|}. \tag{54}$$

6.1. Advection of a cosine bell

The first test case [43] simulates the advection of a cosine bell through one rotation around the sphere over a 12-day period. This test is primarily used to verify positivity and monotonicity and further allows the accuracy of the advective component of the numerical method to be evaluated using standard error norms. The prescribed wind field is non-divergent, and so the continuity equation represents an advection equation for the tracer distribution. The velocity field is prescribed initially and left unmodified by the numerical method.

The initial height field is given by

$$h = \begin{cases} \left(\frac{h_0}{2}\right)\left(1 + \cos\frac{\pi r}{R}\right) & \text{if } r < R, \\ 0 & \text{otherwise,} \end{cases} \tag{55}$$

where r is the great circle distance from the center of the height profile, which is initially located at $(\lambda, \varphi) = (3\pi/2, 0)$. The height of the profile is chosen to be $h_0 = 1000$ m, and its radius is $R = a/3$ (recall that a denotes the radius of the Earth). The non-divergent velocity field is specified in latitude–longitude (φ, λ) coordinates as

$$u_\lambda = u_0(\cos\varphi \cos\tilde{\alpha} + \cos\lambda \sin\varphi \sin\tilde{\alpha}), \tag{56}$$

$$u_\varphi = -u_0 \sin\lambda \sin\tilde{\alpha}, \tag{57}$$

where $u_0 = \pi a/6 \text{ day}^{-1}$. Here, the parameter $\tilde{\alpha}$ denotes the rotation angle transcribed between the physical north pole and the center of the northern panel on the cubed-sphere grid (and should not be confused with the equiangular coordinate α).

The simulation is run for 12 days using $\Delta t = 45$ min (Courant number $= 0.5$) with the unlimited scheme, positivity-preserving limiter, and monotonicity-preserving limiter. We use grid rotation angles $\tilde{\alpha} = 0°$ and $\tilde{\alpha} = 45°$ at a fixed grid resolution of c40. Relative errors are then calculated after 12 days and presented in Table III. For both the positivity-preserving scheme and monotonicity-preserving scheme, the error norms suggest no evidence of undershoots or overshoots after one rotation. Both the unlimited scheme and monotonicity-preserving scheme show reduced accuracy when compared against the positivity-preserving scheme, in the first case because the unlimited scheme experiences increased errors due to numerical oscillations and in the second case because the monotonicity-preserving scheme crops off the maximum of the cosine bell. These errors are apparent when examining contour plots of the absolute error for each method, which are depicted in Figure 5. As with other Eulerian methods on the cubed-sphere grid (such as [14]), we observe that the rotated test (with $\tilde{\alpha} = 45°$) tends to lead to improved error norms over the unrotated test (with $\tilde{\alpha} = 0°$). This result is likely because the velocity field is parallel to coordinate lines for

DOI: 10.1002/fld

Table III. Relative errors in the height field h for advection of a cosine bell at a resolution of c40 and after $t = 12$ days.

Limiting	Direction	L_1 error	L_2 error	L_∞ error	Minimum	Maximum
None	$\tilde{\alpha} = 0°$	5.01 (−2)	3.09 (−2)	2.46 (−2)	−1.50 (−2)	−1.02 (−2)
	$\tilde{\alpha} = 45°$	4.38 (−2)	2.53 (−2)	1.95 (−2)	−1.18 (−2)	−8.44 (−3)
Positive	$\tilde{\alpha} = 0°$	2.84 (−2)	1.75 (−2)	1.56 (−2)	0	−1.01 (−2)
	$\tilde{\alpha} = 45°$	2.61 (−2)	1.52 (−2)	1.31 (−2)	0	−1.03 (−2)
Monotone	$\tilde{\alpha} = 0°$	4.55 (−2)	4.87 (−2)	1.07 (−1)	0	−1.07 (−1)
	$\tilde{\alpha} = 45°$	4.43 (−2)	4.96 (−2)	1.14 (−1)	0	−1.13 (−1)

The value here is represented as 'm $(-b)$' for the sake of readability, which should be read as $m \times 10^{-b}$.

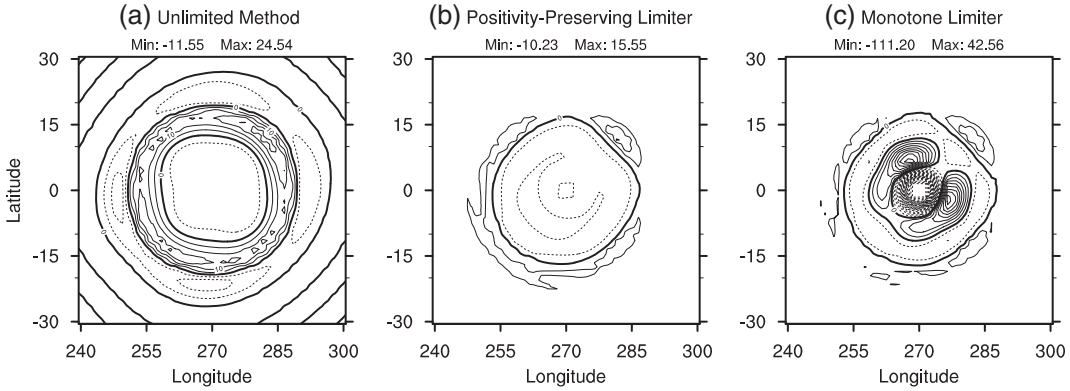

Figure 5. Difference between the numerically computed solution and analytical solution (a) without limiting, (b) with positivity-preserving limiter, and (c) with monotone limiter after one rotation (12 days) on a c40 grid. The direction of rotation is $\tilde{\alpha} = 45°$, which corresponds to motion to the top right. Contours are in intervals of 5 m with solid lines denoting positive contours and dashed lines denoting negative contours. The zero line is enhanced.

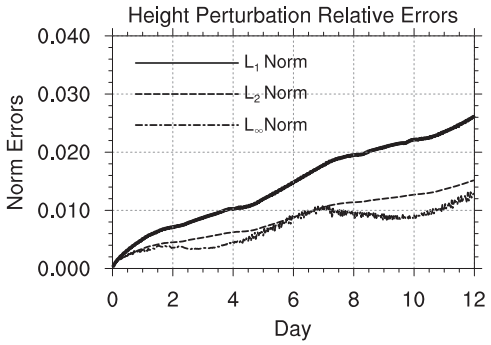

Figure 6. Time series of the normalized height errors for the cosine bell advection with positivity-preserving limiter and direction of rotation $\alpha = 45°$ after one rotation (12 days) on a c40 grid.

approximately half of the total simulation time (as the cosine bell passes over the panel edges). Tests run with alternative choices of time-step size Δt have been performed but do not suggest a strong sensitivity of the method to choice of Δt. The temporal evolution of error norms for this scheme is presented in Figure 6 for $\tilde{\alpha} = 45°$ with the positivity-preserving limiter. No significant sensitivity to the underlying grid is observed in the evolution of these error norms.

To verify convergence of this method with grid refinement, a sequence of tests have been run with $\tilde{\alpha} = 45°$ and grid resolutions c20, c40, c80, and c160. Both the unfiltered and positivity-preserving schemes have been investigated. The time step Δt for each resolution is 90, 45, 22.5,

DOI: 10.1002/fld

Table IV. Relative errors in the height field h for advection of a cosine bell at $t = 12$ days for c20, c40, c80, and c160 resolutions.

Resolution	L_1 error	L_2 error	L_∞ error	Minimum	Maximum
Unfiltered, rotated test case ($\tilde{\alpha} = 45°$)					
c20	2.70 (−1)	1.56 (−1)	1.53 (−1)	−3.46 (−2)	−1.34 (−1)
c40	4.38 (−2)	2.53 (−2)	1.95 (−2)	−1.18 (−2)	−8.44 (−3)
c80	7.32 (−3)	5.31 (−3)	4.90 (−3)	−3.83 (−3)	−1.09 (−3)
c160	1.33 (−3)	1.28 (−3)	1.42 (−3)	−1.31 (−3)	−1.29 (−4)
Order	2.56	2.31	2.22	1.58	3.30
Positivity-preserving, rotated test case ($\tilde{\alpha} = 45°$)					
c20	1.74 (−1)	1.29 (−1)	1.45 (−1)	0	−1.29 (−1)
c40	2.61 (−2)	1.52 (−2)	1.31 (−2)	0	−1.03 (−2)
c80	6.18 (−3)	4.69 (−3)	5.37 (−3)	0	−1.09 (−3)
c160	1.56 (−3)	1.64 (−3)	2.60 (−3)	0	−1.29 (−4)
Order	2.24	2.06	1.87	–	3.31

The computed order of accuracy is obtained from a least-squares fit through the data. The value here is represented as 'm ($-b$)' for the sake of readability, which should be read as $m \times 10^{-b}$.

and 11.25 min, respectively. Error norms for this study are presented in Table IV. Although the scheme is formally third-order accurate, the cosine bell field is only C^1 and so will lead to at most second-order convergence. Consequently, at low resolutions, the scheme exhibits near-third-order convergence, which flattens to second order at higher resolution. The extremum of the cosine bell, which is infinitely smooth, exhibits third-order convergence as expected.

The results from this test compare favorably with other methods, in particular against [14] and [45], keeping in mind that the semi-Lagrangian approach in this paper is formally third-order accurate. Further tests of the advective component of this method can be found in [46], which presents tests from a new standard test case suite for tracer transport [47], including novel mixing diagnostics [48].

6.2. *Steady-state geostrophically balanced flow*

Test case 2 of [43] simulates a zonally symmetric geostrophically balanced flow. This test utilizes an unstable equilibrium solution to the shallow-water equations, which is generally not preserved in atmospheric models that do not use the latitude–longitude grid. However, this test is nonetheless useful to study the convergence properties of numerical methods. The analytical height field is given by

$$h = h_0 - \frac{1}{g}\left(\Omega u_0 a + \frac{u_0^2}{2}\right)\left(-\cos\lambda\cos\varphi\sin\tilde{\alpha} + \sin\varphi\cos\tilde{\alpha}\right)^2, \tag{58}$$

with background height h_0 and velocity amplitude u_0 chosen to be

$$h_0 = \frac{2.94 \times 10^4 \text{ m}^2 \text{ s}^{-2}}{g} \quad \text{and} \quad u_0 = \frac{\pi a}{6} \text{ day}^{-1}. \tag{59}$$

This height field also serves as the reference solution. Again, the parameter $\tilde{\alpha}$ denotes the angle transcribed between the physical north pole and the center of the northern panel. The velocity field is the same as in (56) and (57) and so is not repeated here.

We sample the velocity field pointwise at element nodes and apply high-order Gaussian quadrature to initialize the element-averaged height field. The model is then run for 5 days with a time step of $\Delta t = 3.75$ min at c40 resolution (Courant number = 0.27). The results of the convergence study are given in Table V for the unrotated ($\tilde{\alpha} = 0°$) and rotated ($\tilde{\alpha} = 45°$) grids. As expected, we observe near-third-order convergence for the semi-Lagrangian scheme in both the rotated and unrotated tests, with a slightly better performance from the unrotated test. Plots of the absolute error after 5 days are shown in Figure 7 for the rotated test. As expected, the largest errors associated with

DOI: 10.1002/fld

Table V. Relative errors in the height field h for the steady-state geostrophically balanced flow test at $t = 5$ days.

Resolution	L_1 error	L_2 error	L_∞ error
Unrotated test case ($\tilde{\alpha} = 0°$)			
c20	3.78 (−5)	4.81 (−5)	1.27 (−4)
c40	2.75 (−6)	3.55 (−6)	9.57 (−6)
c80	2.72 (−7)	3.59 (−7)	9.61 (−7)
c160	3.38 (−8)	4.44 (−8)	1.19 (−7)
Order	3.37	3.36	3.35
Rotated test case ($\tilde{\alpha} = 45°$)			
c20	3.24 (−5)	5.39 (−5)	2.08 (−4)
c40	4.44 (−6)	6.56 (−6)	2.36 (−5)
c80	5.11 (−7)	7.67 (−7)	2.78 (−6)
c160	6.48 (−8)	9.57 (−8)	3.45 (−7)
Order	3.00	3.05	3.08

The computed order of accuracy is obtained from a least-squares fit through the data. The value here is represented as 'm $(-b)$' for the sake of readability, which should be read as $m \times 10^{-b}$.

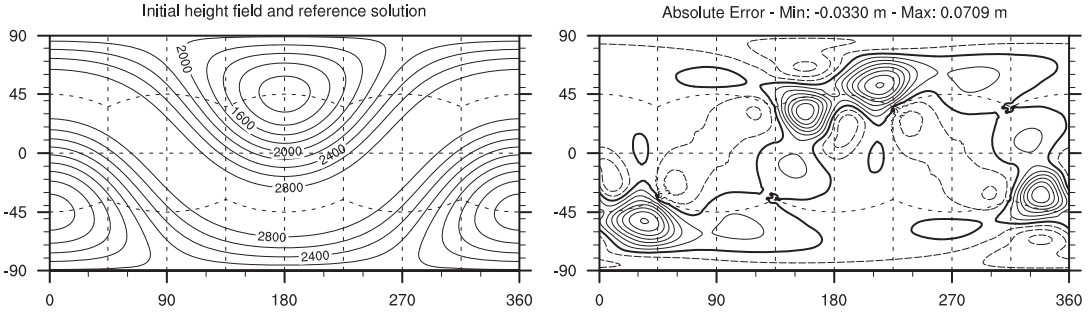

Figure 7. Height field (left, in m) and absolute errors associated with the semi-Lagrangian scheme on a c40 grid for the steady-state geostrophically balanced flow test with $\tilde{\alpha} = 45°$ (right). Contour lines are in units of 0.01 m, with solid lines corresponding to positive values and long dashed lines corresponding to negative values. The thick line corresponds to zero error. The short dashed lines show the location of the underlying cubed-sphere grid.

this test seem to occur in regions where both the gradient of the height field is steepest and the flow field is misaligned with the underlying grid. The error norms for this test compare favorably with other third-order schemes in the literature; in particular, the error norms for the semi-Lagrangian approach appear to be roughly a factor of 5 smaller than those of [14].

6.3. Steady-state geostrophically balanced flow with compact support

Test case 3 of [43] considers another geostrophically balanced flow, but in this case, the velocity field is chosen to be a nonlinearly sheared zonal jet with compact support. This test is particularly difficult for semi-Lagrangian methods because the nonlinear shearing leads to significant deformation of the flux volume. The analytical velocity field is given in rotated latitude–longitude coordinates (φ', λ') by

$$u'_\lambda = u_0 b(x) b(x_e - x) \exp(4/x_e) \quad \text{and} \quad u'_\varphi = 0, \tag{60}$$

where

$$b(x) = \begin{cases} 0 & \text{if } x \leqslant 0 \\ \exp(-1/x) & \text{if } 0 < x \end{cases} \tag{61}$$

DOI: 10.1002/fld

and

$$x = x_e \frac{(\varphi' - \varphi_b)}{(\varphi_e - \varphi_b)}. \tag{62}$$

The grid rotation angle $\tilde{\alpha}$ is imposed using the rotated coordinate system described in [43]. The initial and reference height field is given by

$$h = h_0 - \frac{a}{g} \int_{-\pi/2}^{\varphi'} \left(2\Omega \sin\tau + \frac{u'_\lambda(\tau) \tan\tau}{a} \right) u'_\lambda(\tau) d\tau, \tag{63}$$

which must be integrated numerically at each point where h is required. The background height and velocity amplitude are again chosen to be

$$h_0 = \frac{2.94 \times 10^4 \text{ m}^2 \text{ s}^{-2}}{g} \quad \text{and} \quad u_0 = \frac{\pi a}{6} \text{ day}^{-1}. \tag{64}$$

Further, the compact velocity field has additional free parameters chosen as

$$\varphi_b = -\frac{\pi}{6}, \quad \varphi_e = \frac{\pi}{2}, \quad \text{and} \quad x_e = 0.3. \tag{65}$$

As with the geostrophically balanced flow test described in Section 6.2, this test case represents an unstable equilibrium solution to the shallow-water equations and so is generally not preserved in shallow-water models on non-latitude–longitude grids.

As before, we sample the velocity field pointwise and use high-order Gaussian quadrature to initialize the element-averaged height fields in the numerical model. The test is then run for 5 days, and the final solution is compared against the initial state. The grid rotation angle is chosen as either $\tilde{\alpha} = 0°$ or $\tilde{\alpha} = 60°$. The time step at c40 resolution for this test is $\Delta t = 3.75$ min (Courant number $= 0.2$). The error norms at day 5 obtained for this test for various choices of resolution are given in Table VI. As anticipated, we observe better-than-third-order convergence of this test as resolution is refined for both the rotated and unrotated versions of the test, as errors in the calculation of the upstream velocity field improve with fourth-order accuracy. At coarser resolutions, the rotated version attains slightly smaller errors, likely because for $\tilde{\alpha} = 60°$, the velocity field is more closely aligned with the grid. The reference field and absolute errors for $\tilde{\alpha} = 60°$ are depicted in Figure 8.

Table VI. Relative errors in the height field h for the geostrophically balanced flow with compact support test at $t = 5$ days.

Resolution	L_1 error	L_2 error	L_∞ error
Unrotated test case ($\tilde{\alpha} = 0°$)			
c20	4.74 (−4)	7.50 (−4)	2.53 (−3)
c40	3.24 (−5)	5.39 (−5)	2.08 (−4)
c80	2.02 (−6)	3.44 (−6)	1.37 (−5)
c160	1.68 (−7)	2.90 (−7)	1.16 (−6)
Order	3.84	3.80	3.72
Rotated test case ($\tilde{\alpha} = 60°$)			
c20	2.76 (−4)	4.90 (−4)	2.19 (−3)
c40	1.88 (−5)	3.41 (−5)	1.61 (−4)
c80	1.56 (−6)	2.83 (−6)	1.38 (−5)
c160	1.93 (−7)	3.49 (−7)	1.71 (−6)
Order	3.50	3.50	3.45

The computed order of accuracy is obtained from a least-squares fit through the data. The value here is represented as 'm $(-b)$' for the sake of readability, which should be read as $m \times 10^{-b}$.

DOI: 10.1002/fld

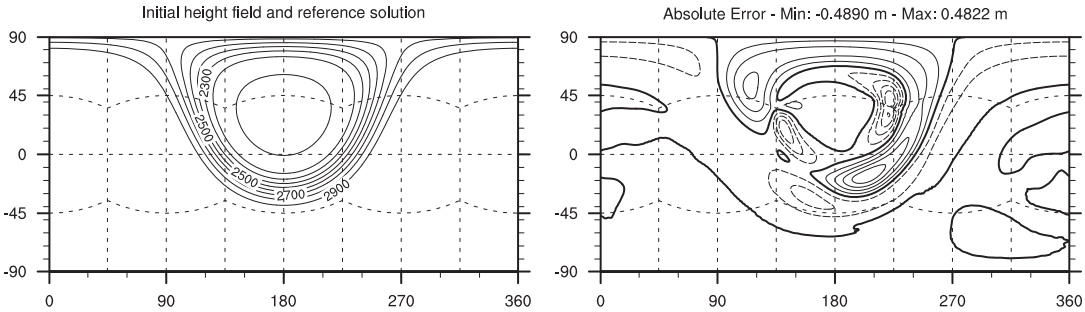

Figure 8. Height field (left, in m) and absolute errors on a c40 grid for the geostrophically balanced flow with compact support test with $\tilde{\alpha} = 60°$ (right). Contour lines are in units of 0.1 m, with solid lines corresponding to positive values and dashed lines corresponding to negative values. The thick line corresponds to zero error.

Again, we see that maximal errors occur in regions where the gradient of the height field is maximal and the flow field is most significantly misaligned with the cubed-sphere grid. At lower resolutions, error norms for this test are roughly comparable with or slightly worse than those reported in [14], because the semi-Lagrangian scheme struggles with strong nonlinearly shearing in the velocity field. However, at higher resolutions, this discrepancy has disappeared, and the semi-Lagrangian scheme again produces better overall results.

6.4. Zonal flow over an isolated mountain

Test case 5 in [43] considers zonal flow with a topographically driven source term. The wind and height fields are defined as in Section 6.2, except with $\tilde{\alpha} = 0°$, $h_0 = 5960$ m and $u_0 = 20$ m s^{-1}. A conical mountain is used for the topographic forcing, given by

$$z = z_0(1 - r/R), \tag{66}$$

with $z_0 = 2000$ m, $R = \pi/9$, and $r^2 = \min[R^2, (\lambda - \lambda_c)^2 + (\varphi - \varphi_c)^2]$. The center of the mountain is at $\lambda_c = 3\pi/2$ and $\varphi_c = \pi/6$. The reference solution for this test is [10], which is run on the spectral transform shallow-water model (STSWM) at a T426 resolution. This high-resolution reference solution was computed by the German Weather Service. The T426 simulation utilizes a Gaussian grid with 640×1280 grid points in latitudinal and longitudinal directions, which corresponds to a grid spacing of about 31 km at the equator. The STSWM results are sampled on a cubed-sphere grid of c40 resolution using high-order Gaussian quadrature for comparison with our simulated element averages.

We simulate this test case on a c40 grid with a time step of 4.25 minutes for 50 days in order to verify the long-term stability of the scheme (initial Courant number = 0.27). The results at days 5, 10, and 15 are plotted in Figure 9. These results agree visually with the numerically computed reference solution of this test. Total energy E, defined by

$$E = \frac{1}{2}h\mathbf{v} \cdot \mathbf{v} + \frac{1}{2}g\left(H^2 - z^2\right), \tag{67}$$

is an invariant quantity in the shallow-water equations. Although some numerical diffusivity is generally required to ensure stability of a numerical scheme, this diffusivity will also lead to loss of energy over time, which is particularly important for long-term climate simulations. In Figure 10, we plot the normalized total energy difference, given by

$$\Delta E(t) = \frac{E(t) - E(t = 0)}{E(t = 0)}, \tag{68}$$

for the first 15 days of this test along with the numerically sampled reference solution. The total energy loss is roughly comparable with the more diffusive third-order scheme of [14] but is generally

DOI: 10.1002/fld

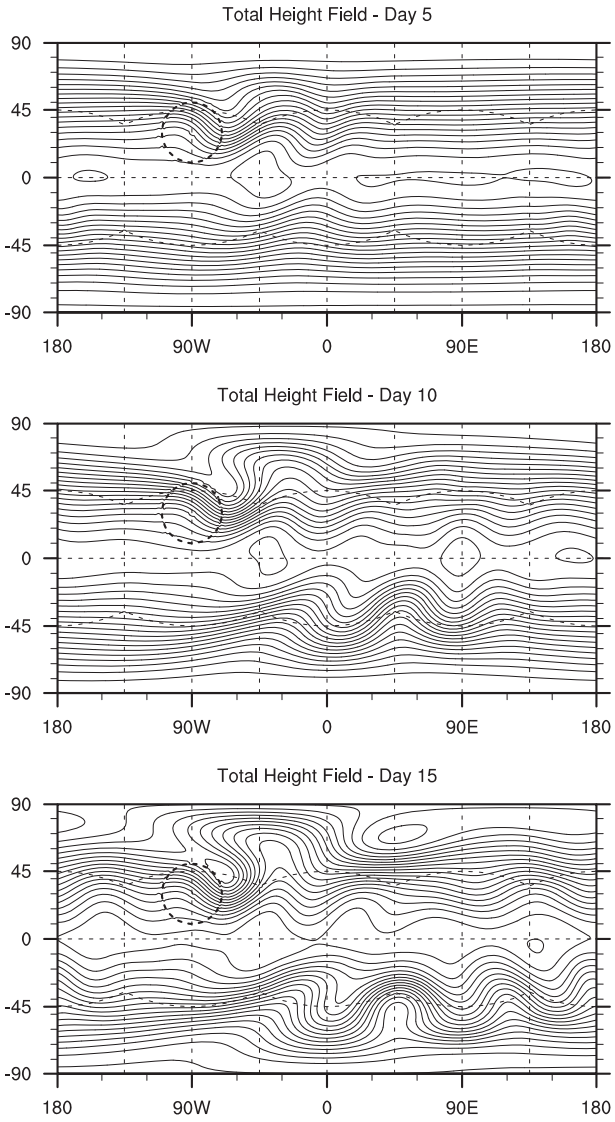

Figure 9. Surface height field H for the flow over an isolated mountain test case at days 5, 10, and 15 on a c40 resolution grid. Contour levels are from 4950 to 5950 m in intervals of 50 m, with the highest elevation being near the equator (the enclosed contours). The dashed circle represents the location of the conical mountain.

still fairly small. We expect that energy conservation will improve with third-order accuracy if the model is run at higher resolutions. For practical applications, where nearly exact conservation of energy is required (such as in 3D simulations), one could augment hyperdiffusion (Section 4.2) with a corresponding source term for internal energy. However, such a modification would still not completely conserve total energy because the semi-Lagrangian advection operator implicitly includes a diffusive averaging mechanism. Hence, for long-term simulations, a global energy fixer would likely be necessary.

6.5. Rossby–Haurwitz wave

Test case 6 in [43] consists of a Rossby-Haurwitz wave with wavenumber 4. This test is an analytical solution of the nonlinear barotropic vorticity equation on the sphere but is not an exact solution of the shallow-water equations. For this test, the height and velocity fields are known analytically at the initial time, but at later times, the solution is only known from a comparison with a high-resolution

DOI: 10.1002/fld

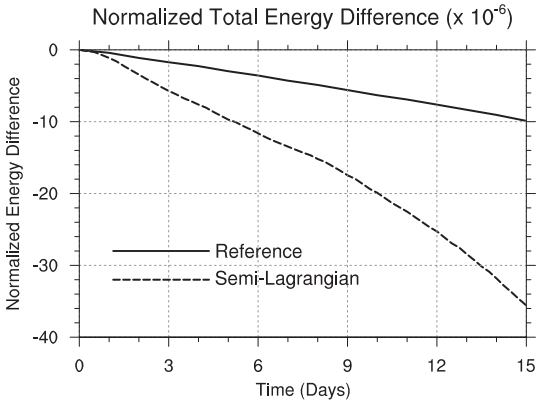

Figure 10. Normalized total energy difference for the flow over an isolated mountain test case at c40 resolution for the semi-Lagrangian scheme and spectral transform shallow-water model reference scheme. Note that the total energy difference has been scaled by a factor of 10^{-6}, so the total energy loss over 15 days represents roughly 0.0036% of total energy.

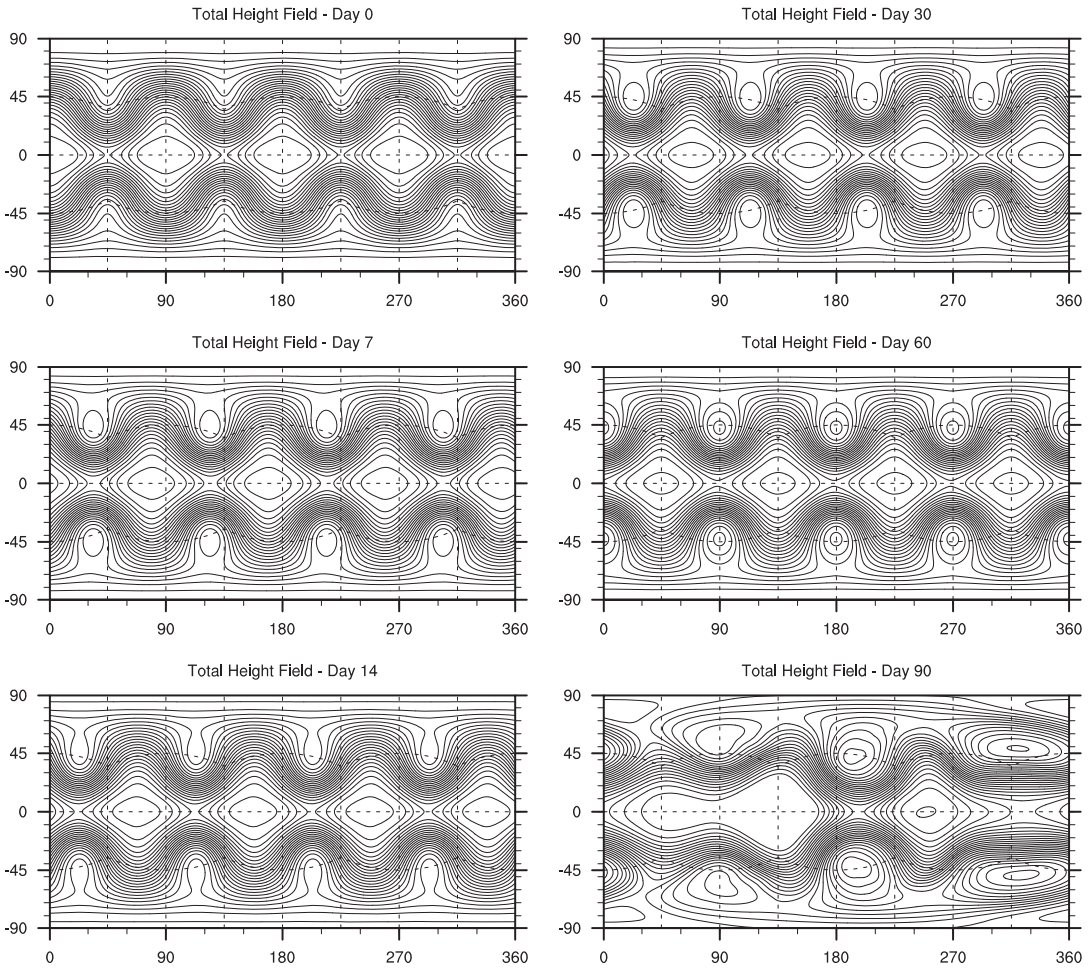

Figure 11. Height field of the wavenumber 4 Rossby–Haurwitz wave. The solution is computed at c80 resolution on days 0, 7, and 14 (left column, from top to bottom) and days 30, 60, and 90 (right column, from top to bottom). The contour levels are from 8100 to 10,500 m in increments of 100 m, with the innermost contours being the highest.

DOI: 10.1002/fld

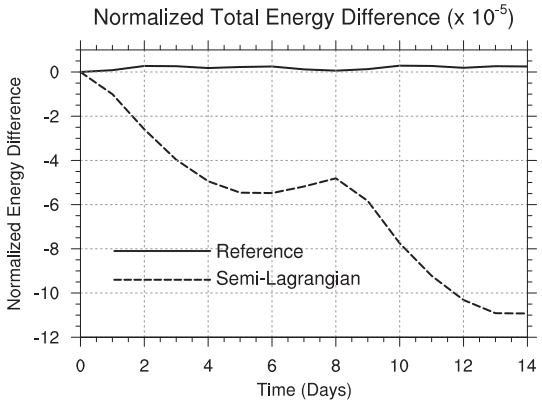

Figure 12. Normalized total energy difference for the Rossby–Haurwitz wave test case simulated on a c40 grid for the semi-Lagrangian scheme and STSWM reference scheme. Note that the total energy difference has been scaled by a factor of 10^{-5}, so the total energy loss over 14 days represents roughly 0.011% of total energy.

numerically computed reference solution. We choose initial fields identical to those specified in [43], which are not repeated here for brevity. Initialization of the height field is via high-order Gaussian quadrature.

It is well known that the wavenumber 4 Rossby–Haurwitz wave is susceptible to instability, which can be driven by truncation errors in the initial conditions (e.g., [49]), and hence, the simulation will generally lose symmetry at some point. The exact time of breakdown is highly dependent on the numerical scheme and does not necessarily depend on the resolution or formal order of accuracy.

We plot the height field for our scheme in Figure 11 at days 0, 7, 14, 30, 60, and 90 at a grid resolution of c80 and using a time step of $\Delta t = 100$ s. The higher resolution required by this test is due to small-scale features in the wave profile that are only captured at this resolution. Observable breakdown of symmetry occurs at roughly 65 days. As in Section 6.4, we calculate the total energy at each time step and plot the normalized total energy difference in Figure 12 for a grid resolution of c40 and time-step size $\Delta t = 200$ s (initial Courant number $= 0.32$) and compare it against the STSWM reference solution (computed at T511 resolution, corresponding to an approximately 26-km resolution). The total loss of energy is again roughly comparable with the results reported in [14].

6.6. *Barotropic instability*

The barotropic instability test case of [44] consists of a zonal jet with compact support at a latitude of 45°, with a latitudinal profile roughly analogous to a much stronger version of test case 3 of [43]. A small height perturbation is added atop the jet, which leads to the controlled formation of an instability in the flow. The relative vorticity of the flow field at day 6 can then be visually compared against a high-resolution numerically computed solution [44, 50]. For comparison, we use the simulation without additional explicit diffusion, because the additional diffusion suggested in [44] leads to a significantly different flow field. Relative vorticity at day 6 obtained from the semi-Lagrangian scheme is plotted for the c40, c80, c120, and c160 resolution grids in Figure 13. These simulations use a time step of $\Delta t = 160$ s at c40 resolution (initial Courant number $= 0.25$), which is scaled downward at higher resolutions to maintain a constant Courant number.

This test case is particularly difficult for models using the cubed sphere to handle [50]. Because the jet is significantly stronger than test case 3 of [43], is aligned in such a way that it passes over cubed-sphere panel edges eight times, and is driven by a relatively mild perturbation, it turns out that wavenumber 4 grid forcing is significant in disturbing the collapse for resolutions less than approximately c100. For higher resolutions, however, we observe rough convergence to the reference solution given by [50] and similarity to the solution calculated by [14].

DOI: 10.1002/fld

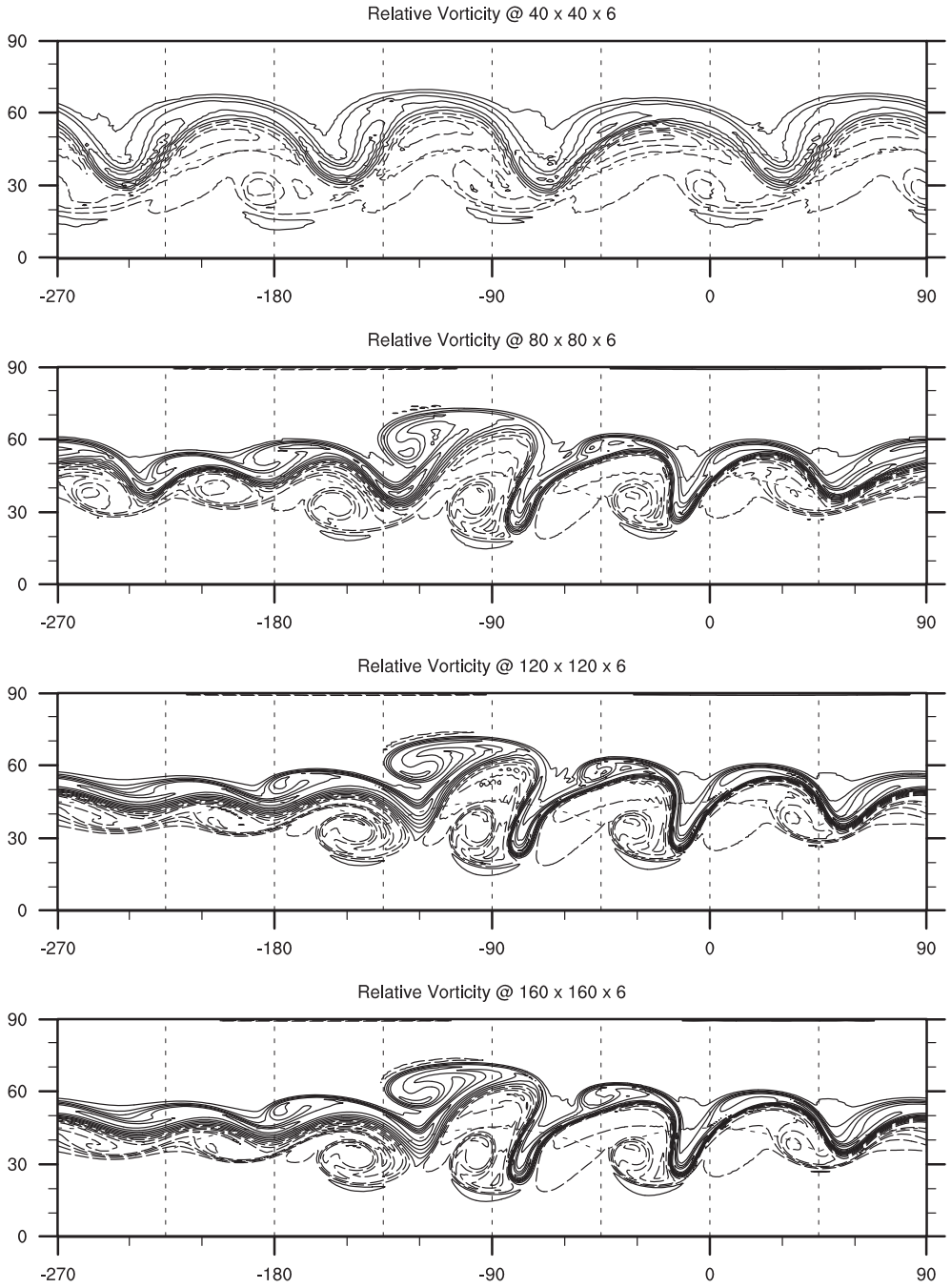

Figure 13. Relative vorticity field associated with the barotropic instability test at day 6 at c40 (top), c80, c120, and c160 resolution (bottom). Contour lines are in increments of $2.0 \times 10^{-5}\ \mathrm{s}^{-1}$ from -1.1×10^{-4} to $-0.1 \times 10^{-4}\ \mathrm{s}^{-1}$ (dashed) and from 0.1×10^{-4} to $1.5 \times 10^{-4}\ \mathrm{s}^{-1}$ (solid). The zero line is omitted. Only the northern hemisphere is depicted in this plot.

6.7. *Efficiency considerations*

A wall-clock comparison has been run for the semi-Lagrangian method (c40 grid with $\Delta t = 300\,\mathrm{s}$) and the high-order finite-volume method of [14] (c40 grid with $\Delta t = 900$ s, three-stage Runge–Kutta operator) on the geostrophically balanced flow test (as in Section 6.2). The total run-time for the serial semi-Lagrangian method was observed to be 30.8 s, compared with the serial

DOI: 10.1002/fld

Table VII. Wall-clock time required for the major components of the semi-Lagrangian method as observed for the geostrophically balanced flow test over a 5-day total integration period ($\Delta t = 300$ s).

Process	Runtime (percent of total)
Update velocities	10.9
Advection	80.6
Compute trajectories	44.4
Compute area-integrated weights	22.1
Reconstruction (one tracer)	4.0
Monotone limiter (one tracer)	1.4
Other overhead (one tracer)	10.1
Other	8.5

Each added tracer would theoretically only need to apply operations indicated by (one tracer).

high-order finite-volume method with 34.5 s. Both codes were run with identical command-line options and optimization flags, although there is inevitably a sensitivity to software implementation which was not accounted for in this test.

A breakdown of the simulation time for the semi-Lagrangian scheme is given in Table VII. The clear majority (80.6%) of computation time is occupied by the advection operator, although only 15.5% of run-time (4.7 s) is needed per tracer. Of the per-tracer requirement, the monotonicity filter adds approximately 0.4 s to the total run-time.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have developed a semi-Lagrangian shallow-water model based on the CSLAM scheme [7, 8, 30] using explicit time-stepping and local flux operators. This model is built on the quasi-uniform cubed-sphere grid so as to maximize parallel performance of the underlying scheme. This scheme has been subjected to a suite of standard test cases, including the test suite of [43] and the barotropic instability [44], to verify accuracy, stability, and convergence. We have confirmed third-order convergence of the scheme with spatial refinement and have generally observed excellent performance as compared with other shallow-water models. Although the strict time-step restrictions for our semi-Lagrangian scheme do not lead to a significant improvement in performance over other models, this approach nonetheless maintains an implicit consistency between the dry air mass and tracer mass fields, which is desirable for applications in atmospheric chemistry.

The work in this paper lays the foundation for a full atmospheric dynamical core using the semi-Lagrangian discretization. In extending this model to a full 3D atmospheric model, we have the option of either using a fully 3D implementation of the CSLAM scheme or using some form of dimension splitting [33]. Work is ongoing to determine which of these approaches may be more desirable for a fully 3D implementation.

APPENDIX A: THE SUB-GRID-SCALE RECONSTRUCTION

In this appendix, we provide our sub-grid-scale reconstruction strategy for each of the mass fields. Our approach leads to a third-order accurate reconstruction, which correctly accounts for the underlying geometry. Monotonicity and positivity are enforced by appropriately limiting the reconstruction prior to the advection step.

A.1. Computing the reconstruction coefficients

The reconstruction strategy proceeds as follows. The stencil we use in the reconstruction step is depicted in Figure A.1. First and second derivatives are calculated using standard finite-difference

DOI: 10.1002/fld

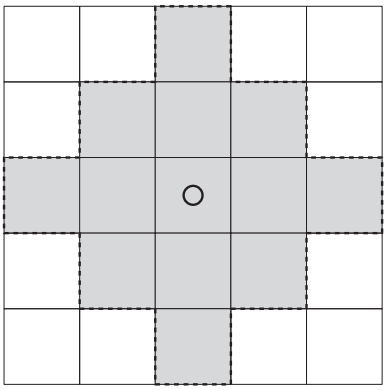

Figure A.1. A depiction of the stencil used for computing the third-order sub-grid-scale reconstruction on the cubed sphere.

formulae, which leads to approximations that are $O(\Delta\alpha^2)$ accurate.

$$D_\alpha \psi_{i,j} = \frac{-\overline{\psi}_{i+2,j} + 8\overline{\psi}_{i+1,j} - 8\overline{\psi}_{i-1,j} + \overline{\psi}_{i-2,j}}{12\Delta\alpha}, \tag{A.1}$$

$$D_\beta \psi_{i,j} = \frac{-\overline{\psi}_{i,j+2} + 8\overline{\psi}_{i,j+1} - 8\overline{\psi}_{i,j-1} + \overline{\psi}_{i,j-2}}{12\Delta\beta}, \tag{A.2}$$

$$D_{\alpha\alpha} \psi_{i,j} = \frac{-\overline{\psi}_{i+2,j} + 16\overline{\psi}_{i+1,j} - 30\overline{\psi}_{i,j} + 16\overline{\psi}_{i-1,j} - \overline{\psi}_{i-2,j}}{24\Delta\alpha^2}, \tag{A.3}$$

$$D_{\alpha\beta} \psi_{i,j} = \frac{\overline{\psi}_{i+1,j+1} - \overline{\psi}_{i-1,j+1} - \overline{\psi}_{i+1,j-1} + \overline{\psi}_{i-1,j-1}}{4\Delta\alpha\Delta\beta}, \tag{A.4}$$

$$D_{\beta\beta} \psi_{i,j} = \frac{-\overline{\psi}_{i,j+2} + 16\overline{\psi}_{i,j+1} - 30\overline{\psi}_{i,j} + 16\overline{\psi}_{i,j-1} - \overline{\psi}_{i,j-2}}{24\Delta\beta^2}. \tag{A.5}$$

A third-order reconstruction relies on obtaining a $O(\Delta\alpha^3)$ approximation to the centerpoint value of ψ, which additionally correctly accounts for the underlying cubed-sphere geometry. Here, we follow the deconvolution procedure of [33], which leads to the following fourth-order approximation in terms of the approximate derivatives of the underlying field and pre-computed derivatives of the cubed-sphere Jacobian:

$$\psi_{(0)i,j} = \overline{\psi}_{i,j} - \frac{\Delta\alpha^4}{12|\mathcal{Z}|_{i,j}} \frac{\partial J}{\partial \alpha} D_\alpha \psi - \frac{\Delta\beta^4}{12|\mathcal{Z}|_{i,j}} \frac{\partial J}{\partial \beta} D_\beta \psi - \frac{\Delta\alpha^2}{24} D_{\alpha\alpha} \psi - \frac{\Delta\beta^2}{24} D_{\beta\beta} \psi. \tag{A.6}$$

Integration over flux volumes is performed in gnomonic coordinates. Hence, the derivatives in equiangular (α, β) coordinates must be converted to the gnomonic basis. We initially convert first derivatives to gnomonic coordinates via

$$D_X \psi_{i,j} = \frac{1}{1 + X^2} D_\alpha \psi_{i,j} \text{ and} \tag{A.7}$$

$$D_Y \psi_{i,j} = \frac{1}{1 + Y^2} D_\beta \psi_{i,j} \tag{A.8}$$

DOI: 10.1002/fld

and then second derivatives using

$$D_{XX}\psi_{i,j} = \frac{1}{1+X^2}\left(-XD_X\psi_{i,j} + \frac{1}{1+X^2}D_{\alpha\alpha}\psi_{i,j}\right), \tag{A.9}$$

$$D_{XY}\psi_{i,j} = \frac{1}{(1+X^2)(1+Y^2)}D_{\alpha\beta}\psi_{i,j}, \text{ and} \tag{A.10}$$

$$D_{YY}\psi_{i,j} = \frac{1}{1+Y^2}\left(-YD_Y\psi_{i,j} + \frac{1}{1+Y^2}D_{\beta\beta}\psi_{i,j}\right). \tag{A.11}$$

Upon computing all gnomonic derivatives, the third-order reconstruction within element $\mathcal{Z}_{i,j}$ takes the form

$$\begin{aligned}\psi_{i,j}(\mathbf{X}) = \psi_{(0)i,j} &+ (X-X_i)D_X\psi_{i,j} + (Y-Y_j)D_Y\psi_{i,j} \\ &+ (X-X_i)^2\frac{D_{XX}\psi_{i,j}}{2} + (X-X_i)(Y-Y_j)D_{XY}\psi_{i,j} + (Y-Y_j)^2\frac{D_{YY}\psi_{i,j}}{2},\end{aligned} \tag{A.12}$$

where $\mathbf{X} = (X, Y)$ is the vector form of the gnomonic coordinate.

The reconstruction coefficients $c_{(p,q)}$, which are then required in the expansion (42), are computed by expanding (A.12) and collecting like terms. This procedure leads to the following set of reconstruction coefficients:

$$\begin{aligned}c_{(0,0)} = \psi_{(0)i,j} &- X_i D_X\psi_{i,j} - Y_j D_Y\psi_{i,j} \\ &+ X_i^2\frac{D_{XX}\psi_{i,j}}{2} + X_iY_jD_{XY}\psi_{i,j} + Y_j^2\frac{D_{YY}\psi_{i,j}}{2},\end{aligned} \tag{A.13}$$

$$c_{(1,0)} = D_X\psi_{i,j} - X_iD_{XX}\psi_{i,j} - Y_jD_{XY}\psi_{i,j}, \tag{A.14}$$

$$c_{(0,1)} = D_Y\psi_{i,j} - Y_jD_{YY}\psi_{i,j} - X_iD_{XY}\psi_{i,j}, \tag{A.15}$$

$$c_{(2,0)} = \frac{D_{XX}\psi_{i,j}}{2}, \tag{A.16}$$

$$c_{(1,1)} = D_{XY}\psi_{i,j}, \tag{A.17}$$

$$c_{(0,2)} = \frac{D_{YY}\psi_{i,j}}{2}. \tag{A.18}$$

A.2. Limiter procedure

The advection algorithm currently supports two limiters. A positivity-preserving limiter is available to avoid spurious negative values due to undershoots in the reconstruction, and a stricter monotonic limiter is available for removing all unphysical oscillations. The limiters follow the approach of [51], wherein extreme values of the sub-grid-scale reconstruction are detected and the reconstruction is scaled so that these extreme values fit within some predefined range. For the positivity-preserving limiter, the range is simply chosen to be $[0, +\infty]$, implying maximum values of the reconstruction are left untouched while minimum values are cropped to zero if they are anywhere negative. For the monotonicity-preserving limiter, the range is chosen to be $[\phi_{\min}, \phi_{\max}]$, where

DOI: 10.1002/fld

$$\phi_{\min} = \min_{\mathcal{N}_k} \overline{\phi}_k,$$

$$\phi_{\max} = \max_{\mathcal{N}_k} \overline{\phi}_k,$$

and $\mathcal{N}_k$ is the set of all neighboring elements to element k, including element k itself. On a regular Cartesian grid, even diagonal neighbors are considered when determining the minimum and maximum values of the scalar field, so in total nine elements are used.

A.3. Treatment of panel edges

Panel edges in cubed-sphere geometry require some additional consideration because they represent discontinuities in the equiangular coordinate system. To compute central derivatives, as required for both the sub-grid-scale reconstruction of the height field and derivatives in the velocity evolution, we must extend each panel outward into a 'halo region', which overlaps neighboring panels. A remapping scheme is then required to remap the height field and velocity field into the halo region using only known information on neighboring panels.

The remapping scheme we use for the semi-Lagrangian dynamical core is identical to the method discussed in [14] for element-averaged scalar fields. This approach first builds the sub-grid-scale reconstruction in neighboring elements using one-sided derivatives and then samples the resulting reconstruction at Gaussian quadrature points in order to assemble a fourth-order approximation to the element average of each scalar field in the halo region. For the velocity field, we instead use a fourth-order one-sided sampling scheme to sample the velocity field at nodal points in the halo region, as we only require knowledge of pointwise values. This approach has been shown to be effective at suppressing low-order errors due to the coordinate discontinuity.

APPENDIX B: UPSTREAM INTEGRATION OF THE MASS FLUX

The mass flux is computed in gnomonic coordinates, with volume element $dV = J_{XY}(\mathbf{X})dXdY$. Consequently, (41) can be written for an arbitrary edge as

$$F = \sigma \int_{\mathbf{a}_k^\tau} \psi(\mathbf{X}) J_{XY}(\mathbf{X}) dXdY, \tag{B.1}$$

where $\sigma \in \{-1, 1\}$ is again a sign indicator depending on the direction of the flux and $\mathbf{a}_k^\tau$ denotes an arbitrary flux region. Summation over distinct flux regions is implied, as in cases 3 and 4 of Figure 4. The Jacobian in gnomonic coordinates is given by J_{XY}, which takes the form

$$J_{XY}(\mathbf{X}) = \frac{1}{(1 + X^2 + Y^2)^{3/2}}. \tag{B.2}$$

With an appropriate sub-grid-scale reconstruction of the form (42), we can write (B.1) as

$$F = \sigma \sum_{p,q} c_{(p,q)} \int_{\mathbf{a}_k^\tau} X^p Y^q J_{XY}(\mathbf{X}) dXdY \tag{B.3}$$

and so reduce the problem to a linear combination of the reconstruction coefficients $c_{(p,q)}$ and the integrated Jacobian-weighted polynomial basis functions. This formulation is particularly efficient when transporting multiple tracers, as the integrals must only be computed once for each pair (p, q).

DOI: 10.1002/fld

To compute the geometric integrals (B.3), we must first introduce a coordinate transform that maps the flux region to the unit square. Here, we describe the procedure for an edge of constant X, noting that the equations for an edge of constant Y are analogous. The coordinates $\mathbf{s} = (s,t) \in [0,1]^2$ are defined implicitly via

$$X(\mathbf{s}) = X(s,t) = X_1 + s\left(a^2 t + bt + c\right) \quad \text{and} \quad Y(\mathbf{s}) = Y(s,t) = Y_1 + fs + (s(g-f) + \Delta X)t, \tag{B.4}$$

where $f = Y_1' - Y_1$, $g = Y_3' - Y_3$ and $\Delta X = Y_3 - Y_1$. The quadratic coefficients (a,b,c) are defined by fitting a quadratic through points

$$(t,X) = \left\{(0, X_1' - X_1), (t_*, X_2' - X_1), (1, X_3' - X_1)\right\}, \quad \text{with} \quad t_* = \frac{Y_2' - Y_1'}{Y_3' - Y_1'}. \tag{B.5}$$

This choice leads to

$$a = \frac{X_3' t_* - X_2' + X_1'(1-t_*)}{t_*(1-t_*)}, \tag{B.6}$$

$$b = \frac{X_2' - X_1' + t_*^2\left(X_1' - X_3'\right)}{t_*(1-t_*)}, \tag{B.7}$$

$$c = X_1' - X_1. \tag{B.8}$$

This construction requires that $Y_1' \neq Y_3'$, $Y_3' \neq Y_2'$, and $Y_1' \neq Y_2'$, which should not occur for sufficiently laminar flows and small enough Courant number. These coordinate axes are depicted for a certain generalized quadrilateral in Figure B.1.

Integration of (B.3) by substitution then leads to an additional Jacobian term that takes the form

$$\Phi(\mathbf{s}) = \left|\det\left(\frac{\partial(X,Y)}{\partial(s,t)}\right)\right| = \left|(-a(g-f)t^2 - 2aft - bf + c(g-f))s + \frac{(X(\mathbf{s}) - X_1)\Delta X}{s}\right|. \tag{B.9}$$

Consequently, the geometric integrals can be written as

$$\int_{\mathbf{a}_k^\tau} X^p\, Y^q\, J_{XY}(\mathbf{X})\, dX dY = \int_{s=0}^{1}\int_{t=0}^{1} X^p\, Y^q\, J_{XY}(\mathbf{X}(\mathbf{s}))\, \Phi(\mathbf{s}) dt ds. \tag{B.10}$$

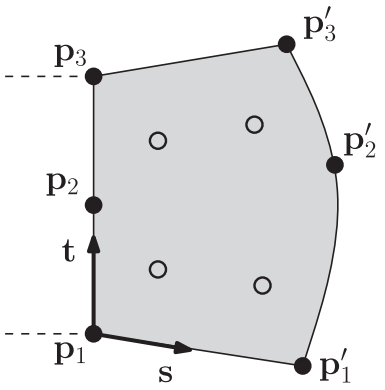

Figure B.1. The location of quadrature points (empty circles) for an arbitrary generalized quadrilateral and associated coordinate axes (s,t).

DOI: 10.1002/fld

Given an arbitrary quadrature rule with quadrature points $\mathbf{s}_k = (s_k, t_k)$ and associated weights w_k, the numerical integral is then computed via

$$\int_{\mathbf{a}_k^\tau} X^p Y^q J_{XY}(\mathbf{X}) dX dY = \sum_k X(\mathbf{s}_k)^p Y(\mathbf{s}_k)^q J_{XY}(\mathbf{X}(\mathbf{s}_k)) \Phi(\mathbf{s}_k) w_k. \tag{B.11}$$

For quadrilateral integration, we use a four-point fourth-order quadrature rule (Figure B.1) given by

$$\mathbf{s}_1 = \left(-\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right), \quad \mathbf{s}_2 = \left(\frac{1}{\sqrt{3}}, -\frac{1}{\sqrt{3}}\right), \quad \mathbf{s}_3 = \left(-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right), \quad \mathbf{s}_4 = \left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right), \tag{B.12}$$

and $w_i = 1/4$ for each $i \in \{1, 2, 3, 4\}$.

ACKNOWLEDGEMENTS

Support for this work has been provided by the University of California Davis and the Office of Science, US Department of Energy, award no. DE-SC0003990. The National Center for Atmospheric Research is sponsored by the National Science Foundation.

REFERENCES

1. Neale RB, Chen C-C, Gettelman A, Lauritzen PH, Park S, Williamson DL, Conley AJ, Garcia R, Kinnison D, Lamarque J-F, Marsh D, Mills M, Smith AK, Tilmes S, Vitt F, Cameron-Smith P, Collins WD, Iacono MJ, Easter RC, Liu X, Ghan SJ, Rasch PJ, Taylor MA. Description of the NCAR Community Atmosphere Model (CAM 5.0). *Technical Report NCAR Technical Note NCAR/TN-486+STR*, National Center for Atmospheric Research: Boulder, Colorado, June 2010. Available from: http://www.cesm.ucar.edu/models/cesm1.0/cam/, [Accessed on 5 September 2013].
2. Dennis JM, Edwards J, Evans KJ, Guba O, Lauritzen PH, Mirin AA, St-Cyr A, Taylor MA, Worley PH. CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model. *International Journal of High Performance Computing Applications* 2012; **26**(1):74–89.
3. Taylor M, Tribbia J, Iskandarani M. The spectral element method for the shallow water equations on the sphere. *Journal of Computational Physics* 1997; **130**:92–108.
4. Donner LJ, Wyman BL, Hemler RS, Horowitz LW, Ming Y, Zhao M, Golaz J-C, Ginoux P, Lin S-J, Schwarzkopf MD, Austin J, Alaka G, Cooke WF, Delworth TL, Freidenreich SM, Gordon CT, Griffies SM, Held IM, Hurlin WJ, Klein SA, Knutson TR, Langenhorst AR, Lee H-C, Lin Y, Magi BI, Malyshev SL, Milly PCD, Naik V, Nath MJ, Pincus R, Ploshay JJ, Ramaswamy V, Seman CJ, Shevliakova E, Sirutis JJ, Stern WF, Stouffer RJ, Wilson RJ, Winton M, Wittenberg AT, Zeng F. The dynamical core, physical parameterizations, and basic simulation characteristics of the atmospheric component AM3 of the GFDL global coupled model CM3. *Journal of Climate* 2011; **24**(13):3484–3519.
5. Putman WM, Lin SJ. A finite-volume dynamical core on the cubed-sphere grid. In *Numerical Modeling of Space Plasma Flows: Astronum-2008*, Vol. 406, Pogorelov NV, Audit E, Colella P, Zank GP (eds). Astronomical Society of the Pacific Conference Series, 2009; 268–276.
6. Putman WM, Lin SJ. Finite-volume transport on various cubed-sphere grids. *Journal of Computational Physics* 2007; **227**:55–78.
7. Lauritzen PH, Nair RD, Ullrich PA. A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid. *Journal of Computational Physics* 2010; **229**:1401–1424.
8. Harris LM, Lauritzen PH, Mittal R. A flux-form version of the conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed sphere grid. *Journal of Computational Physics* February 2011; **230**:1215–1237.
9. Lauritzen PH, Ullrich PA, Nair RD. Atmospheric transport schemes: desirable properties and a semi-Lagrangian view on finite-volume discretizations. In *Numerical Techniques for Global Atmospheric Models, Lecture Notes in Computational Science and Engineering*, Vol. 80. Springer: Berlin, 2011; 185–250.
10. Jakob-Chien R, Hack JJ, Williamson DL. Spectral transform solutions to the shallow water test set. *Journal of Computational Physics* 1995; **119**:164–187.
11. Heikes R, Randall DA. Numerical integration of the shallow water equations on a twisted icosahedral grid. Part I: basic design and results of tests. *Monthly Weather Review* 1995; **123**:1862–1880.
12. Ronchi C, Iacono R, Paolucci PS. The "cubed sphere": a new method for the solution of partial differential equations in spherical geometry. *Journal of Computational Physics* 1996; **124**(1):93–114.
13. Ringler TD, Jacobsen D, Gunzburger M, Ju L, Duda M, Skamarock WB. Exploring a multiresolution modeling approach within the shallow-water equations. *Monthly Weather Review* 2011; **139**:3348–3368.

DOI: 10.1002/fld

14. Ullrich PA, Jablonowski C, van Leer BL. High-order finite-volume models for the shallow-water equations on the sphere. *Journal of Computational Physics* 2010; **229**:6104–6134.
15. Lin S-J, Rood RB. An explicit flux-form semi-Lagrangian shallow water model on the sphere. *Quarterly Journal of the Royal Meteorological Society* 1997; **123**:2477–2498.
16. Chen C, Xiao F. Shallow water model on cubed-sphere by multi-moment finite volume method. *Journal of Computational Physics* 2008; **227**:5019–5044.
17. Côté J, Staniforth A. An accurate and efficient finite-element global model of the shallow-water equations. *Monthly Weather Review* 1990; **118**:2707–2717.
18. Thomas SJ, Loft RD. The NCAR spectral element climate dynamical core: semi-implicit Eulerian formulation. *Journal of Scientific Computing* 2005; **25**(1):307–322. DOI: http://dx.doi.org/10.1007/s10915-004-4646-2.
19. Giraldo FX, Hesthaven JS, Wartburton T. Nodal high-order discontinuous Galerkin methods for the shallow water equations. *Journal of Computational Physics* 2002; **181**:499–525.
20. Nair RD, Thomas SJ, Loft RD. A discontinuous Galerkin global shallow water model. *Monthly Weather Review* 2005; **133**:876–888.
21. Wong M, Skamarock WC, Lauritzen PH, Stull RB. A cell-integrated semi-Lagrangian semi-implicit shallow-water model (CSLAM-SW) with conservative and consistent transport. *Monthly Weather Review* 2013; **141**:2545–2560.
22. Kwizak M, Robert AJ. A semi-implicit scheme for grid point atmospheric models of the primitive equations. *Monthly Weather Review* 1971; **99**(1):32–36.
23. Clancy C, Lynch P. Laplace transform integration of the shallow water equations. Part 2: Lagrangian formulation and orographic resonance. *Quarterly Journal of the Royal Meteorological Society* 2011; **137**:800–809.
24. Laprise JP, Plante A. A class of semi-lagrangian integrated-mass (SLM) numerical transport algorithms. *Monthly Weather Review* 1995; **123**(2):553–565.
25. Leslie LM, Purser RJ. Three-dimensional mass-conserving semi-Lagrangian scheme employing forward trajectories. *Monthly Weather Review* 1995; **123**(8):2551–2566.
26. Zerroukat M, Wood N, Staniforth A. SLICE: a semi-Lagrangian inherently conserving and efficient scheme for transport problems. *Quarterly Journal of the Royal Meteorological Society* 2002; **128**(586):2801–2820.
27. Zerroukat M, Wood N, Staniforth A. SLICE-S: a semi-Lagrangian inherently conserving and efficient scheme for transport problems on the sphere. *Quarterly Journal of the Royal Meteorological Society* 2004; **130**(602):2649–2664.
28. Leonard BP, Lock AP, MacVean MK. Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Monthly Weather Review* 1996; **124**(11):2588–2606.
29. Lin SJ, Rood RB. Multidimensional flux-form semi-Lagrangian transport schemes. *Monthly Weather Review* 1996; **124**:2046.
30. Ullrich PA, Lauritzen PH, Jablonowski C. Some considerations for high-order 'incremental remap'-based transport schemes: edges, reconstructions, and area integration. *International Journal for Numerical Methods in Fluids* 2012; **71**:1131–1151.
31. Taylor MA, Edwards J, St.Cyr A. Petascale atmospheric models for the community climate system model: new developments and evaluation of scalable dynamical cores. *Journal of Physics: Conference Series* 2008; **125**:012023.
32. Sadourny R. Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids. *Monthly Weather Review* 1972; **100**:136–144.
33. Ullrich PA, Jablonowski C. MCore: a non-hydrostatic atmospheric dynamical core utilizing high-order finite-volume methods. *Journal of Computational Physics* 2012; **231**:5078–5108.
34. Arakawa A, Lamb VR. Computational design and the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics* 1977; **17**:173–265.
35. Randall DA. Geostrophic adjustment and the finite-difference shallow-water equations. *Monthly Weather Review* 1994; **122**:1371. DOI: 10.1175/1520-0493(1994)122⟨1371:GAATFD⟩2.0.CO;2.
36. Rusanov VV. Calculation of intersection of non-steady shock waves with obstacles. *Journal of Computational Mathematics & Physics USSR* 1961; **1**:267–279.
37. Hundsdorfer W, Ruuth SJ, Spiteri RJ. Monotonicity-preserving linear multistep methods. *SIAM Journal on Numerical Analysis* 2003; **41**(2):605–623.
38. Hirt C, Amsden A, Cook J. An arbitrary Lagrangian–Eulerian computing method for all flow speeds. *Journal of Computational Physics* 1974; **14**(3):227–253.
39. Lauritzen PH, Erath C, Mittal R. On simplifying 'incremental remap'-based transport schemes. *Journal of Computational Physics* 2011; **230**:7957–7963.
40. Cossette JF, Smolarkiewicz PK. A Monge–Ampére enhancement for semi-Lagrangian methods. *Computers & Fluids* 2011; **46**(1):180–185.
41. Ullrich PA, Lauritzen PH, Jablonowski C. Geometrically exact conservative remapping (GECoRe): regular latitude–longitude and cubed-sphere grids. *Monthly Weather Review* 2009; **137**:1721–1741.
42. Zalesak ST. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics* June 1979; **31**:335–362.
43. Williamson DL, Drake JB, Hack JJ, Jakob R, Swarztrauber PN. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics* 1992; **102**:211–224.
44. Galewsky J, Scott RK, Polvani LM. An initial-value problem for testing numerical models of the global shallow-water equations. *Tellus Series A* 2004; **56**:429–440.
45. Rossmanith JA. A wave propagation method for hyperbolic systems on the sphere. *Journal of Computational Physics* 2006; **213**:629–658.

46. Lauritzen PH, Ullrich PA, Jablonowski C, Andronova N, Bosler PA, Calhoun D, Enomoto T, Dong L, Dubey S, Guba O, Hansen AB, Juang HMH, Kaas E, Kent J, Muller R, Penner JE, Prather MJ, Reinert D, Skamarock WC, Sorensen B, Taylor MA, White JB. A standard test case suite for two-dimensional linear transport on the sphere: results from a collection of state-of-the-art schemes. *Geoscientific Model Development* 2013; **6**:4983–5076.
47. Lauritzen PH, Skamarock WB, Prather MJ, Taylor MA. A standard test case suite for two-dimensional linear transport on the sphere. *Geoscientific Model Development* 2011; **5**:189–228.
48. Lauritzen PH, Thuburn J. Evaluating advection/transport schemes using interrelated tracers, scatter plots and numerical mixing diagnostics. *Quarterly Journal of the Royal Meteorological Society* 2012; **138**(665):906–918.
49. Thuburn J, Li Y. Numerical simulations of Rossby Haurwitz waves. *Tellus Series A* 2000; **52**:181–189.
50. St-Cyr A, Jablonowski C, Dennis JM, Tufo HM, Thomas SJ. A comparison of two shallow-water models with nonconforming adaptive grids. *Monthly Weather Review* 2008; **136**:1898–1922.
51. Barth TJ, Jespersen DC. The design and application of upwind schemes on unstructured meshes. *Proceedings of AIAA 27th Aerospace Sciences Meeting*, Reno, Nevada, January 9–12, 1989. AIAA Paper 89-0366. 13p.

DOI: 10.1002/fld