# Efficient Conservative Global Transport Schemes for Climate and Atmospheric Chemistry Models

Ramachandran D. Nair

*Department of Marine, Earth and Atmospheric Sciences and Department of Mathematics, North Carolina State University, Raleigh, North Carolina*

Jeffrey S. Scroggs

*Department of Mathematics, North Carolina State University, Raleigh, North Carolina*

Fredrick H. M. Semazzi

*Department of Marine, Earth and Atmospheric Sciences and Department of Mathematics, North Carolina State University, Raleigh, North Carolina*

ABSTRACT

A computationally efficient mass-conservative transport scheme over the sphere is proposed and tested. The scheme combines a conservative finite-volume method with an efficient semi-Lagrangian scheme based on the dimension splitting "cascade" method. In the regions near the poles where the conservative cascade procedure breaks down, a globally conservative, but locally approximate scheme is used. This procedure is currently restricted to polar meridional Courant numbers less than one. The resulting conservative cascade scheme is evaluated using a solid-body rotation test and deformational flow test, and found to be both accurate and efficient. Compared to the traditional semi-Lagrangian scheme employing a bicubic-Lagrange interpolator, the proposed scheme is considerably more accurate and almost twice as fast while conserving mass exactly.

## 1. Introduction

Current climate models require efficient schemes for transport of humidity, liquid and solid water variables, and a number of chemical constituents. Monotonic versions of semi-Lagrangian (SL) transport schemes have proved to be an efficient numerical method for treating the advection process in climate and atmospheric chemistry models (Williamson and Olson 1998). However, a serious disadvantage of most SL schemes is that they do not formally conserve integral invariants as total mass or total energy (Laprise and Plante 1995; Machenhauer and Olk 1997). The total mass, in particular, has been found to drift significantly if no corrections are applied during longer integrations of the SL climate model (Moorthi et al. 1995). A posteriori correction is often employed to enforce conservation, but may require ad hoc mass restoration or correction (Priestley 1993). Such a posteriori mass-fixing after upstream interpola-

tion is not only computationally expensive but also lacks rigor.

In SL methods, it is customary to use backward trajectories (Staniforth and Côté 1991) with arrival points at regular (Eulerian) grid points. The departure (Lagrangian) points are, in general, not distributed uniformly in the domain. Backward-trajectory SL schemes calculate the values at upstream Lagrangian points from the surrounding values at regular grid points by means of interpolation. These values are the new estimates at a future time level. This idea can be extended for finite-volume based conservative schemes. Such schemes are based on grid cells rather than grid points. Finite-volume based SL schemes have gained prominence during the recent years (e.g., Rančić 1992; Laprise and Plante 1995; Scroggs and Semazzi 1995; Leslie and Purser 1995). However, only a few conservative transport schemes are available for spherical geometry (global) application (e.g., Prather 1986; Smolarkiewicz and Rasch 1991; Rasch 1994), and (most of) such schemes are computationally expensive. Hourdin and Armenguad (1999) have reviewed various finite-volume schemes for global atmospheric advection.

Recent developments in global conservative transport

*Corresponding author address:* Dr. Jeffrey S. Scroggs, Department of Mathematics, North Carolina State University, Campus Box 8205, Raleigh, NC 27695-8205.
E-mail: scroggs@unity.ncsu.edu

schemes include the methods developed by Li and Chang (1996), Lin and Rood (1996), Machenhauer and Olk (1996), and Rasch (1998). Recently, Nair and Machenhauer (2002), hereafter referred to as NM02, generalized the conservative global advection scheme introduced by Machenhauer and Olk (1998). As implemented, this scheme was restricted to meridional Courant numbers less than 2 near the poles. Among the changes to the original Machenhauer and Olk (1998) scheme, NM02 introduced a new formulation in the singular Lagrangian latitude rows, which removed any Courant number restrictions. The main drawbacks of this approach, at least in the formulation of NM02, are that it requires storage of global information for the 2D remapping algorithm used and its computational efficiency is comparable to traditional SL schemes. The conservative scheme presented in this paper does not require storage of global information; furthermore, the conservative remapping is based on the computationally efficient dimension splitting "cascade methods," introduced for interpolation in an SL scheme by Purser and Leslie (1991).

In this paper, a global conservative advection scheme is developed by combining a computationally efficient SL scheme with a mass-conservative 1D finite-volume method, and utilizing the cascade interpolation framework. The cascade interpolation replaces computationally expensive multidimensional interpolation associated with the SL scheme by an efficient sequence of 1D interpolations. For example, a typical tensor product interpolation in 3D, with the basic 1D Lagrange interpolator, requires $O(p^3)$ operations per grid point per field, where $p$ is the formal order of accuracy of the interpolator. However, a 3D cascade scheme can use the same 1D interpolator three times, and needs only $O(p)$ operations, a significant saving (Purser and Leslie 1991; Nair et al. 1999a,b).

Cascade interpolation is applicable to both backward- and forward-trajectory SL schemes (Purser and Leslie 1991). Nair et al. (1999b), significantly reduced the computational overhead of cascade interpolation by simplifying the intermediate grid point calculations. Nair et al. (1999a), hereafter referred to as NCS99, further extended the cascade interpolation scheme for spherical geometry application. The remarkable computational efficiency of the cascade scheme is exploited for designing efficient conservative SL schemes in the new method presented here.

Leslie and Purser (1995) and Rančić (1995), hereafter referred to as R95, have developed mass-conserving SL schemes (forward trajectory) in Cartesian geometry, based on cascade interpolation. However, their approaches have serious limitations for conventional spherical geometry. As in the present paper, R95 combines the accurate 1D finite-volume remapping method based on the piecewise parabolic method of Colella and Woodward (1984), with the computationally efficient cascade interpolation procedure of Purser and Leslie

(1991). However, the Lagrangian cells were not explicitly defined, and as in the scheme of Leslie and Purser (1995) the values of the mass is assigned at the nodes (or "mass points") of the Lagrangian grid. In both schemes, these values were transferred to the regular Eulerian grid using their respective cascade scheme. On the surface of the sphere with a uniform latitude–longitude grid, the area of the cells changes with latitude and an application of both schemes is not obvious. Another difficulty applying to both schemes at the pole arises because some Lagrangian cells (i.e., cells bounded by Lagrangian grid lines) may not be well defined; thus in these regions, the conservative option of cascade interpolation procedure may break down.

The conservative cascade scheme presented in this paper uses explicitly defined intermediate grid cells. This provides a complete geometric basis for the transport. The cell-averaged density is transferred cell to cell by employing 1D conservative remapping through this intermediate grid cell system. For example, in 2D, two 1D conservative remapping operations are required through an intermediate grid cell system to complete the transport.

The change in grid cell area with latitude is taken into account by performing the remappings in a Cartesian ($\lambda$, $\mu = \sin\theta$) coordinate system, where $\lambda$ is the longitude and $\theta$ is the latitude. This idea was introduced by Machenhauer and Olk (1996) and was used in Machenhauer and Olk (1998) and in NM02. In the region near the poles where the conservative cascade procedure breaks down, we use the globally conservative but locally approximate scheme like the one used in NM02. This procedure is currently restricted to polar meridional Courant numbers less than one.

The layout for the remainder of the paper is as follows. In section 2, the formulation of conservative semi-Lagrangian advection and the basic 1D finite-volume representation of grid cells for cascade remapping is described. The computational procedure for intermediate grid generation and the conservative cascade scheme are introduced in section 3. Details of extension to spherical geometry and special treatment for polar caps are also discussed in section 3. Section 4 deals with the numerical experiments using the global conservative cascade scheme, followed by a concluding discussion in section 5.

## 2. Conservative semi-Lagrangian advection

To introduce the basic idea of a conservative Lagrangian scheme, consider the Eulerian flux form of mass continuity and tracer conservation equations in the absence of source or sink,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0, \tag{1}$$

$$\frac{\partial \rho q}{\partial t} + \nabla \cdot (\rho q \mathbf{V}) = 0, \tag{2}$$

where $\rho$ is the density of the fluid, $q$ is the passive scalar (e.g., mixing ratio) per unit mass and $\mathbf{V}$ is the three-dimensional flow velocity. From (1) and (2), in the absence of discontinuities, the Lagrangian form of the advection equation

$$\frac{Dq}{Dt} = 0 \qquad (3)$$

can be derived, where $D/Dt \equiv \partial/\partial t + \mathbf{V} \cdot \mathbf{\nabla}$ is the total (Lagrangian) derivative. However, a semi-Lagrangian discretization of (3) cannot, in general, guarantee the mass-conservation or other properties such as tracer monotonicity.

To formulate the conservative scheme, consider the integral form (Chorin and Marsden 1993) of Eq. (1),

$$\frac{d}{dt} \int_{A(t)} \rho \, dV = 0. \qquad (4)$$

Here, the integral is over $A(t)$, an arbitrary reference volume of fluid (moving with the fluid), and $dV$ is the element of volume. Similarly for Eq. (2), the integral form is

$$\frac{d}{dt} \int_{A(t)} \rho q \, dV = 0. \qquad (5)$$

The properties of integral forms (4) and (5) are discussed in Dukowicz and Baumgardner (2000). Since the solution procedure for these equations are similar, we focus our attention on (4). The integral in Eq. (4) may be interpreted as the *mass* of a quantity in the reference volume to be advected along a trajectory. Discretization of (4) can lead to a mass-conservative advection scheme (Laprise and Plante 1995; Dukowicz and Baumgardner 2000). Moreover, Eq. (4) is suitable for a computationally efficient semi-Lagrangian time integration scheme.

### a. Semi-Lagrangian formulation

The method is described in one spatial dimension, then extended to multiple dimensions. Let $x_{j+1/2}, j = 0, 1, 2, \ldots$ be the regularly spaced Eulerian grid points with grid spacing $\Delta x_j = x_{j+1/2} - x_{j-1/2}$, and let $t^n$, $n = 0, 1, \ldots$ be the discrete times at which we will compute the solution. For a single time step, we define the *departure point* $x^*_{j+1/2}$ at time $t^n$ to be the location of material originally located at $x_{j+1/2}$ at time $t^{n+1}$. That is, the departure point is obtained by setting $x^*_{j+1/2} = B(t^n)$, where $B$ is obtained by solving the ordinary differential equation

$$\frac{dB(t)}{dt} = u(B(t), t), \qquad B(t^{n+1}) = x_{j+1/2} \qquad (6)$$

backward in time. Here $u(x, t)$ is the 1D velocity. The semi-Lagrangian time discretization of (4) in 1D leads to

$$\int_{A(t^{n+1})} \rho \, dx = \int_{A(t^n)} \rho \, dx. \qquad (7)$$

The Eulerian and Lagrangian grid cells may be defined in terms of the Eulerian and Lagrangian grid points, respectively. Here, the $j$th Eulerian cell is defined by the grid points $x_{j-1/2}$ and $x_{j+1/2}$ as the corner points such that the cell width is $\Delta x_j$. The corresponding Lagrangian cell width is $\Delta x^*_j$. Note that, the scalar field (density) to be advected is cell centered and its initial value is a cell-averaged quantity. The grid points formally define each cell, and the values at these points can be estimated from the cell averages.

Stability and conservation of finite-volume based conservative SL methods (hereafter referred to as cell-based methods) require choosing the time step $\Delta t$ in such a way that the trajectories do not cross in one time step (Machenhauer and Olk 1997). This is analogous to the Lipschitz criterion $\|\partial u/\partial x\|\Delta t \leq 1$ of traditional SL advection schemes (Smolarkiewicz and Pudykiewicz 1992). Moreover, this condition implies that the line segments forming a cell do not cross as we traverse trajectories backward in time to the departure points (Scroggs and Semazzi 1995). This is a criterion for the cell to be "well-defined," that is, cell walls should not intersect nor should the cell collapse at a single time step (Lin and Rood 1996; NM02).

The computational domain is assumed to be spanned by nonoverlapping and well-defined Lagrangian cells (except at the polar regions, a topic covered later in this paper). Thus, the cell-based method reduces to find the mass in the Lagrangian cells using (7), that is, estimating mass from the known Eulerian cells intercepted by the Lagrangian cell. Now, representation of scalar fields in the Eulerian cells is discussed.

### b. Representation of density distribution

Cell-based methods can be constructed using a series of cell-averaged quantities rather than discrete point values. The conservative cascade scheme (CCS) uses cell averages to construct a unique piecewise parabolic polynomial for each dependent variable within a cell. The area beneath each curve represents the amount of mass, energy, or other fluid property. A judicious representation was provided by Colella and Woodward (1984), the piecewise parabolic method (PPM). Apart from conservation, other desirable properties such as monotonicity and positive-definiteness can be further satisfied by the polynomials. The PPM has been utilized in meteorological modeling [see, Carpenter et al. (1990); Rančić (1992, 1995); Laprise and Plante (1995); Lin and Rood (1996); Machenhauer and Olk (1996, 1998); NM02], but our new scheme is the first application in the global cascade framework.

Piecewise parabolic functions represent the density distribution in the Eulerian cells. Specifically, when $x \in [x_{j-1/2}, x_{j+1/2}]$, the density for the $j$th Eulerian cell $h^n_j(x)$ is defined to be

$$h^n_j(x) = a_0 + a_1 x + a_2 x^2, \qquad (8)$$

where the dependency of the coefficients $a_0$, $a_1$, and $a_2$ on $j$ and $n$ has been suppressed. For efficient implementation, these coefficients are modified by introducing a normalized local variable (Carpenter et al. 1990),

$$\xi = \frac{x - x_{j-1/2}}{\Delta x_j} - \frac{1}{2},$$

for each cell such that $\xi \in [-1/2, 1/2]$. Thus, the parabolic function (8) representing the density distribution in the $j$th cell can be written as,

$$h_j^n(\xi) = \overline{h}_j^n + \delta h \xi + h_S\left[\frac{1}{12} - \xi^2\right], \qquad (9)$$

where $\overline{h}_j^n$ is the cell-averaged density,

$$\delta h = h_R - h_L, \qquad h_S = 6\overline{h}_j^n - 3(h_L + h_R),$$

and $h_L = h_j^n(-1/2)$ and $h_R = h_j^n(1/2)$ are the left and right cell boundary (edge) values, respectively. The cell-averaged density (or simply average density) $\overline{h}^j$, is defined as,

$$\overline{h}_j^n = \frac{1}{\Delta x_j} \int_{x_{j-1/2}}^{x_{j+1/2}} h_j^n(x)\, dx = \int_{-1/2}^{+1/2} h_j^n(\xi)\, d\xi. \quad (10)$$

Using the definition of average density, the discrete analogue of the conservation Eq. (7) is

$$\overline{h}_j^{n+1} \Delta x_j = \overline{h}_j^* \Delta x_j^*, \qquad (11)$$

where

$$\overline{h}_j^* = \frac{1}{\Delta x_j^*} \int_{x_{j-1/2}^*}^{x_{j+1/2}^*} h^n(x)\, dx, \qquad (12)$$

is the average density in the departure cell and $\Delta x_j^* = x_{j+1/2}^* - x_{j-1/2}^*$ is the width of the departure cell respectively. Thus the scheme reduces to determine the new cell average $\overline{h}_j^{n+1}$ at time $t^{n+1}$, from Eq. (11) using the values at time $t_n$.

The PPM uses cubic interpolation of $\overline{h}_j^n$ (including values in neighboring cells) to obtain the cell boundary values $h_L$ and $h_R$ (Colella and Woodward 1984). Moreover, modifications of the coefficients similar to Carpenter et al. (1990) can be made to create a monotonic or positive-definite version.

### c. Conservative remapping scheme

The calculation of the average density, $\overline{h}_j^{n+1}$, is begun by calculating the mass enclosed in the upstream Lagrangian cell $[x_{j-1/2}^*, x_{j+1/2}^*]$ at the departure time $t^n$. The mass is [cf. Eq. (12)]

$$\overline{h}_j^* \Delta x_j^* = \int_{x_{j-1/2}^*}^{x_{j+1/2}^*} h^n(x)\, dx$$

$$= \int_{x_r}^{x_{j+1/2}^*} h^n(x)\, dx - \int_{x_r}^{x_{j-1/2}^*} h^n(x)\, dx, \quad (13)$$

where $x_r$ is a common reference grid point on the grid

line such that $x_r \leq x_{j-1/2}^*$. The Eq. (13) is convenient for numerical evaluation. The mass integrated from $x_r$ to a Lagrangian point $x_{j-1/2}^*$ is the *accumulated mass* (or cumulative mass) towards the *wall* of the Lagrangian cell at the point $x_{j-1/2}^*$, and can be computed as follows,

$$\int_{x_r}^{x_{j-1/2}^*} h^n(x)\, dx = \sum_{l=1}^{k-1} \overline{h}_l^n \Delta x_l + \int_{x_{k-1/2}}^{x_{j-1/2}^*} h_k^n(x)\, dx, \quad (14)$$

where the reference point, $x_r$, may be chosen as the first grid point from the left, and $x_{j-1/2}^* \in [x_{k-1/2}, x_{k+1/2}]$. The mass in the Lagrangian cell is the accumulated mass towards the left wall subtracted from the accumulated mass towards the right wall of the Lagrangian cell. Thus, the remapping process consists of the determination of accumulated mass for each Lagrangian cell wall using (14). The average density at new time level is given by (11).

A schematic illustration of the 1D remapping considered here is given in Fig. 1 of NM02, also, the same 1D remapping scheme will be used as a basic component of the 2D CCS remapping in the following section. This remapping scheme has been used by R95, Machenhauer and Olk (1996, 1998), NM02 and others for 1D advection. The cumulative mass approach used is similar to the one introduced by Dukowicz (1984) for remapping.

## 3. The global conservative cascade scheme

In order to introduce the basic geometry of the CCS in 2D, consider an Eulerian grid system (orthogonal Cartesian coordinates), defined by the $m \times n$ grid points $(\lambda_i, \mu_j)$ with grid spacing $\Delta \lambda_i$ and $\Delta \mu_j$ in $\lambda$ and $\mu$ directions, respectively. For every grid point $(\lambda_i, \mu_j)$, there exists a unique upstream Lagrangian point $(\lambda_{ij}, \mu_{ij})$ in the domain. Four neighboring Eulerian grid points on the vertices of a rectangular region bounded by Eulerian grid lines defines an Eulerian cell, and the corresponding exact Lagrangian cell is a region whose *corner* points are the Lagrangian points. Figure 1 shows such a $(\lambda, \mu)$-grid system, where the Eulerian points are the grid intersections and the Lagrangian points are marked by filled circles. Consider the Lagrangian cell *ABCD* shown in Fig. 1 with the corner points, $(\lambda_{ij}, \mu_{ij})$, $(\lambda_{i+1,j}, \mu_{i+1,j})$, $(\lambda_{i+1,j+1}, \mu_{i+1,j+1})$, and $(\lambda_{i,j+1}, \mu_{i,j+1})$, respectively (our notation implies a small Courant number, however, the methods does not have this limitation). In Fig. 1, horizontal curves are Lagrangian grid lines corresponding to horizontal Eulerian grid lines. Hereafter, the deformed Lagrangian grid lines are referred to as the Lagrangian latitudes. The vertical grid lines ($\lambda = \lambda_i$) shown in Fig. 1 are Eulerian longitudes. Intersections of the Lagrangian latitudes and the Eulerian longitudes create a hybrid grid system and the resulting grid points are defined as the intermediate grid points. In Fig. 1, the intermediate points are marked with $\times$.

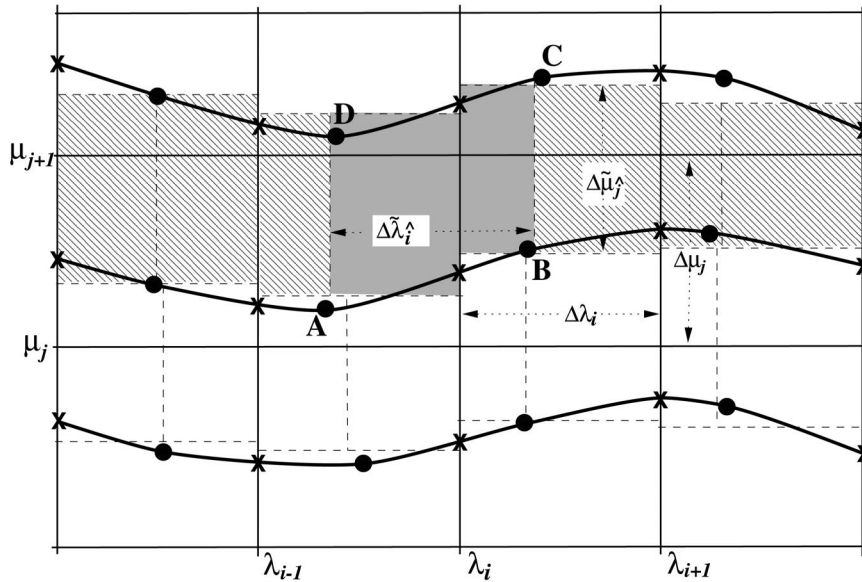The computational procedure is first described for a

FIG. 1. Schematic illustration of the cascade remapping scheme in 2D, where the filled circles represent the Lagrangian grid points corresponding to the Eulerian grid points $(\lambda_i, \mu_j)$. The curves represent Lagrangian latitudes, and their intersections with the vertical grid lines $(\lambda = \lambda_i)$ are the intermediate grid points, denoted by $\times$. The crosshatched rectangular regions show the intermediate cells. An approximated Lagrangian cell (computational cell) corresponding to the Lagrangian cell *ABCD* is shown as the shaded polygon, whose sides are parallel to the Eulerian grid lines.

$(\lambda, \mu)$ Cartesian plane and then extended for spherical geometry.

### a. Computational procedure

The CCS consists of approximating mass enclosed in the Lagrangian cells by two conservative 1D remappings along each of the coordinate directions, respectively. For the computational procedure, the Lagrangian cells are approximated as polygons. The generation of an intermediate grid is a key component of the CCS, because the local conservation and efficiency of the scheme depend upon the grid generation. First, we describe the intermediate grid generation followed by the approximation of the intermediate and Lagrangian cells used for the conservative remapping.

#### 1) INTERMEDIATE GRID GENERATION

Let $(\lambda_\ell, \hat{\mu}_{\ell j})$ be the point formed by the intersection of a Lagrangian latitude and an Eulerian longitude. For example, in Fig. 1, consider the Lagrangian latitude passing through the Lagrangian points $A(\lambda_{ij}, \mu_{ij})$ and $B(\lambda_{i+1,j}, \mu_{i+1,j})$, and the Eulerian longitude $\lambda = \lambda_i$. Then the intermediate grid point lies on segment *AB* of the Lagrangian latitude such that $\lambda_\ell = \lambda_i \in [\lambda_{ij}, \lambda_{i+1,j}]$. However, the value of $\hat{\mu}_{ij}$ interpolated from the neighboring $\mu$ coordinates of *A* and *B*. A linear interpolation method for generating intermediate points (Nair et al. 1999b) is an obvious choice, however, this approach may not be sufficiently accurate for the nonuniform grid

considered here. To compute the location of the intermediate point, we employ a cubic-Lagrange interpolation, requiring four Lagrangian points along the Lagrangian latitude for source points. Thus, to find $(\lambda_i, \hat{\mu}_{ij})$, Lagrangian points $(\lambda_{i-1,j}, \mu_{i-1,j})$ and $(\lambda_{i+2,j}, \mu_{i+2,j})$ to the immediate left of *A* and right of *B,* respectively, (see Fig. 1), can be used. By performing a 1D cubic interpolation with $\{\lambda_{i-1,j}, \lambda_{ij}, \lambda_{i+1,j}, \lambda_{i+2,j}\}$ as the nodes of interpolation, provides the value of $\hat{\mu}_{ij}$ corresponding to the value $\lambda_\ell = \lambda_i$. The collection of all such $(\lambda_\ell, \hat{\mu}_{\ell j})$ uniquely defines an intermediate grid system.

#### 2) INTERMEDIATE AND LAGRANGIAN CELLS

For computational efficiency, Lagrangian latitudes are approximated by lines drawn parallel to the Eulerian $\lambda$ axis. In Fig. 1, these are shown as dashed horizontal lines. In the *i*th column, the south wall of the *j*th intermediate cell is defined to be $\tilde{\mu}_j = (\hat{\mu}_{ij} + \hat{\mu}_{i+1,j})/2$. Similarly, the north cell wall of the intermediate cell $\tilde{\mu}_{j+1}$ is defined. The cell width in the $\mu$ direction of the intermediate cell is $\Delta\tilde{\mu}_j = \tilde{\mu}_{j+1} - \tilde{\mu}_j$. Thus, the intermediate cells are rectangles and they fall in the columns of the Eulerian cells. In Fig. 1, an array of intermediate cells corresponding to the region bounded by two adjacent Lagrangian latitudes is shown as the crosshatched contiguous rectangles. Note that the intermediate cells span the entire computational domain without overlapping or holes.

Now, the Lagrangian cells may be approximated in terms of rectangular regions carved from adjoining in-

termediate cells. These approximated Lagrangian cells are hereafter referred to as the *computational* cells. Each computational cell is defined as a union of nonoverlapping adjacent rectangles from the contiguous intermediate cells. In Fig. 1, a computational cell corresponding to the Lagrangian cell *ABCD* is schematically depicted by the shaded polygon located among an array of intermediate cells (crosshatched regions). The east and west boundary walls of the computational cell are defined as the straight line segments parallel to the $\mu$ axis. For example, the west wall of the computational cell as shown in Fig. 1 (shaded polygon) lies on the vertical line $\tilde{\lambda}_i = (\lambda_{ij} + \lambda_{i,j+1})/2$. This approximates the curve joining *AD*. Similarly, the east wall of the computational cell is defined, and the horizontal width of the computational cell is $\Delta \tilde{\lambda}_i = \tilde{\lambda}_{i+1} - \tilde{\lambda}_i$. Note that, the north and south walls of the computational cells, respectively, are defined by the segments of the north and south walls of the intermediate cells. With this construction, the computational cells are polygons with sides parallel to Eulerian grid lines. The idea to approximate Lagrangian cells by polygons with sides parallel to either $\lambda$ or $\mu$ Eulerian grid lines was originally introduced in Machenhauer and Olk (1998) and was then later used in NM02 to increase the efficiency of their 2D remapping.

### 3) CONSERVATIVE CASCADE REMAPPING

For our 2D scheme, a cascade of two 1D remappings are performed in a consistent way so that the mass transfer is conservative. In the present context, a 1D remapping in $\mu$ direction followed by another 1D remapping in $\lambda$ direction is briefly described as follows with the aid of Fig. 1.

The 1D remapping in $\mu$ direction transfers mass from the Eulerian cell to the intermediate cell. Consider the *i*th column in Fig. 1, let the initial mass in the *j*th Eulerian cell be $M_{ij} = \overline{\rho}_{ij}^n \Delta \mu_j \Delta \lambda_i$, where $\overline{\rho}_{ij}^n$ is the given cell-averaged density at time $t^n$, and $\Delta \mu_j \Delta \lambda_i$ is the cell area. The first step for the conservative remapping is calculation of the average density $\overline{h}_j$ from $M_{ij}$ such that $\overline{h}_j = M_{ij}/\Delta \mu_j$, for each cell in the *i*th column. Here, we have suppressed the dependency of $\overline{h}_j$ on $i$. Next, 1D piecewise parabolic profiles are constructed along the vertical column, utilizing $\overline{h}_j$ as the cell-averaged density for each 1D cell having cell width $\Delta \mu_j$. The mass in the intermediate cell (here it may be logically considered as a 1D cell with cell width $\Delta \tilde{\mu}_j$) is determined using the relation (11) combined with (13). This remapping scheme is employed for each column in the domain, and the mass enclosed in the intermediate cells are determined.

Remapping in $\lambda$ direction is performed in an analogous manner, where the mass from the intermediate cells are transferred to the computational cells. Consider a horizontal array of intermediate cells between two adjacent Lagrangian latitudes (as shown as the crosshatched cells in Fig. 1). The width in the $\lambda$ direction of

the intermediate cell is $\Delta \lambda_i$ and that of the computational cell is $\Delta \tilde{\lambda}_i$. In order to perform the 1D remapping in $\lambda$ direction, the average density per $\Delta \lambda_i$ is computed for each of the intermediate cells from the known values of mass, as described in the case of 1D remapping in $\mu$ direction. Piecewise parabolic profiles are then fitted along the rows in such a way that the area below the parabolas represent the mass. The mass in the computational cell having cell width $\Delta \tilde{\lambda}_i$ is computed using the relation (11) combined with (13), as the difference of the accumulated mass from a reference wall. Thus, the new average density $\overline{\rho}_{ij}^{n+1}$ at the arrival cell can be determined from the known mass, since the area of the arrival cell is already known, using Eq. (7).

The cascade procedure described above can be summarized as follows:

- Generate an intermediate grid system by finding the intersection of Lagrangian latitudes and Eulerian longitudes.
- Determine intermediate and computational cells by determining the midpoint of the east–west intermediate points and north–south Lagrangian points.
- Perform a 1D conservative remapping in $\mu$ direction to transfer mass in the Eulerian cell to the intermediate cell.
- Perform a 1D conservative remapping along the $\lambda$ direction to transfer mass in the intermediate cells to the computational cells. This step completes the cascade cycle.

### b. Extension to spherical geometry

A major difficulty in extending cell-based methods to spherical geometry is the convergence of meridians at the polar singularities of the spherical coordinate system. A special treatment is often needed to obviate the pole problems (e.g., Rasch 1994; Li and Chang 1996; Lin and Rood 1996). Here we restrict our attention to the cases where the polar meridional Courant number $C_\theta \leq 1$ (i.e., the Lagrangian pole point is located within or on the first latitude circle from the North or South Pole).

It is convenient to introduce an area-preserving transformation from the latitude–longitude $(\lambda, \theta)$ grid with $\lambda \in [0, 2\pi]$, and $\theta \in [-\pi/2, +\pi/2]$, to the $(\lambda, \mu)$ grid system with $\mu = \sin\theta$. The variable $\mu$ can be treated as an independent variable such that $\mu \in [-1, 1]$. Under this transformation, the global integral for any arbitrary function $f$ over the sphere (with radius $a = 1$) takes the following form

$$\int_0^{2\pi} \int_{-\pi/2}^{+\pi/2} f(\lambda, \theta) \cos\theta \, d\lambda \, d\theta$$
$$= \int_0^{2\pi} \int_{-1}^{+1} f(\lambda, \mu) \, d\lambda \, d\mu. \tag{15}$$

The resulting $(\lambda, \mu)$ coordinate system may be considered similar to the Cartesian form with north and south poles as grid lines, and periodic in the east–west direction. This application of a Cartesian $(\lambda, \mu)$ coordinate system for conservative remapping on the sphere was originally introduced by Machenhauer and Olk (1996) and was then later used in Machenhauer and Olk (1998) and in NM02. The $(\lambda, \mu)$ coordinates introduced here have nonuniform resolution in the $\mu$ direction. Note that in the $(\lambda, \mu)$ grid system, $\Sigma_j \Delta\mu_j = 2 = \Sigma_j \Delta\tilde{\mu}_j$ and $\Sigma_i \Delta\lambda_i = 2\pi = \Sigma_i \Delta\tilde{\lambda}_i$, where $\Delta\tilde{\mu}_j$ and $\Delta\tilde{\lambda}_i$, respectively, are the grid spacing for the intermediate cell in the $\mu$ direction and that for the computational cell in $\lambda$ direction.

## c. Intermediate grid generation over the sphere

The intermediate grid points can be computed by using the cubic interpolation method described in the previous section. However, a second method based on a great-circle approach (NCS99) is more computationally efficient, but is less accurate. In NCS99, an exact method for the determination of intermediate grid intersections (for nonconservative SL schemes) over the sphere is given. However, this approach may not be suitable for the present conservative cascade scheme since this method does not isolate polar regions containing the singularity. We use a cascade cycle that isolates the polar regions.

In most SL global models, 3D Cartesian coordinates $(X, Y, Z)$ equivalent to the spherical polar coordinates $(\lambda, \theta)$ are computed for solving trajectory equations analogous to Eq. (6) (Côté et al. 1998). In NCS99, Lagrangian longitudes and Eulerian latitude circles are used for determining intermediate intersection, utilizing the $(X, Y, Z)$ coordinates. However, for the present application, intermediate points are defined as the intersection of Lagrangian latitudes with Eulerian longitudes. The advantage of this definition is that the Lagrangian pole cap (that includes the singular point) can be isolated by a region with known exterior boundary.

The intermediate grid point $(\lambda_\ell, \hat{\theta}_{\ell j})$ in spherical coordinate can be determined by using the great-circle approach (NCS99), and then transformed into computational domain such that $(\lambda_\ell, \hat{\mu}_{\ell j} = \sin\hat{\theta}_{\ell j})$. The great-circle plane that passes through the two points $(\lambda_{ij}, \theta_{ij})$ and $(\lambda_{i+1,j}, \theta_{i+1,j})$ is given by

$$\hat{\alpha}X + \hat{\beta}Y + \hat{\gamma}Z = 0, \quad (16)$$

where the formulas for coefficients $(\hat{\alpha}, \hat{\beta}, \hat{\gamma})$ can be found in NCS99. The trigonometric equivalent to (16) is

$$\hat{\alpha}\cos\theta\cos\lambda + \hat{\beta}\cos\theta\sin\lambda + \hat{\gamma}\sin\theta = 0. \quad (17)$$

From (17), the value $\hat{\theta}_{\ell j}$ corresponding to $\lambda_\ell$ of the intermediate point $(\lambda_\ell, \hat{\theta}_{\ell,j})$ that lies on the Lagrangian latitude can be determined by using the following equation,

$$\hat{\theta}_{\ell,j} = \tan^{-1}\left[\frac{-(\hat{\alpha}\cos\lambda_\ell + \hat{\beta}\sin\lambda_\ell)}{\hat{\gamma}}\right]. \quad (18)$$

Thus, a set of points $(\lambda_\ell, \hat{\theta}_{\ell,j})$ [or equivalently $(\lambda_\ell, \hat{\mu}_{\ell,j})$] defines an intermediate grid system over the surface of the sphere or on the $(\lambda, \mu)$ plane. The cell-to-cell mass transfer is performed through the intermediate grid during the cascade cycle.

## d. Special treatment for polar caps

Since the Lagrangian cell which includes the singular point (Eulerian pole) is not well defined in the $(\lambda, \mu)$ plane, a special treatment is needed for determination of mass in such *singular cells*. Here, a polar cap is a region on the surface of the sphere that includes the pole point. The exterior boundary of the Eulerian polar cap is a latitude circle; in the $(\lambda, \mu)$ plane, this corresponds to a rectangular region. Both the exterior boundary of the Lagrangian polar cap and the intermediate polar cap, by design, are the same deformed Lagrangian latitude. In the $(\lambda, \mu)$ plane, for the northern hemisphere, this is a region bounded by the lines $\mu = 1$, $\lambda = 0$, $\lambda = 2\pi$ and a curve marking the south boundary (see NM02 for relevant illustrations). Thus, the mass enclosed in the intermediate polar cap is the same as that in the Lagrangian polar cap. This means that the total mass $T^m$, enclosed in the Lagrangian polar cap is computed during the first sweep of conservative remapping in $\mu$ direction and does not change in the second phase. Therefore, $T^m$ can be redistributed to each of the constituent computational cells in the Lagrangian polar cap. Only the total mass enclosed in the Lagrangian pole cap is needed during the cascade cycle; thus, the east and west walls of the computational cells in this zone are approximated as straight line segments parallel to the $\mu$ axis, and joining the south boundary and the grid line $\mu = 1$. Now, the redistribution method remains to be discussed.

The total mass $T^m$ in the Lagrangian polar cap is redistributed to each of the constituent cells by using the method considered by NM02. To find the *weights* for redistribution, a regular SL scheme combined with a cubic-Lagrange interpolation is used (NM02). Lagrangian cell centers are treated as upstream points and the values of the mass in the cells are considered as the advecting field. Then the values $m_k$ at the upstream positions are estimated with cubic-Lagrange interpolation, and then used for constructing weights of mass redistribution. The weight is defined to be

$$w_k = \frac{|m_k|}{\sum_k |m_k|}. \quad (19)$$

Computed mass in the constituent Lagrangian cell is then $C_k^m = T^m w_k$, such that $\Sigma_k C_k^m = T^m$ (NM02), and leading to a conservative scheme.

The accuracy near the pole may be improved by introducing additional points in the $\mu$ direction on the east

and west walls. Thus, a series of Lagrangian cells along the latitude circle on the polar zones, hereafter referred to as *Lagrangian belt* (NM02), may be subdivided in $\mu$ direction at a known ratio, by linearly interpolating the Cartesian coordinates $(X, Y, Z)$ of the cell wall's north and south corner points. The values of $(\lambda, \mu)$ at the newly introduced positions are calculated from the Cartesian coordinates. This avoids computing additional trajectories for the new points (NM02).

## 4. Numerical experiments

The computational efficiency, accuracy, and robustness of the algorithm presented earlier are demonstrated through a series of advection experiments that utilize solid-body rotation and deformational flow. A scalar field $\psi$, that may be considered as density or tracer distribution, is advected.

For the present study, the Lagrangian belts enclosing the Lagrangian polar cap have been subdivided in the $\mu$ direction as described previously. In the north polar zone, the belt surrounding the Lagrangian polar cap has been subdivided into three sub-belts. This is done by introducing two sets of points on the east–west walls around the latitude. Similarly, the belt next to it (south) has been subdivided into two sub-belts. Also in the south polar zone, new sets of sub-belts have been symmetrically introduced.

### a. Solid-body rotation test

The details of the solid-body rotation experiment are in Williamson et al. (1992). The parameters of the experiment used here are the same as NCS99, with velocity components

$$u = u_0(\cos\alpha \, \cos\theta + \sin\alpha \, \cos\lambda \, \sin\theta), \quad (20)$$

$$v = -u_0 \, \sin\alpha \, \sin\lambda, \quad (21)$$

where $\alpha$ is the angle between the axis of solid-body rotation and the polar axis of the spherical coordinate system (Williamson et al. 1992). When $\alpha = 0$, the axis of rotation is the polar axis, and when $\alpha = \pi/2$ it is in the equatorial plane. The initial scalar distribution is the cosine bell,

$$\psi(\lambda, \theta) = \begin{cases} 1/2[1 + \cos(\pi r/R)], & \text{if } r < R \\ 0, & \text{if } r \geq R \end{cases} \quad (22)$$

where,

$$r = \cos^{-1}[\sin\theta_c \, \sin\theta + \cos\theta_c \, \cos\theta \, \cos(\lambda - \lambda_c)], \quad (23)$$

is the great-circle distance between $(\lambda, \theta)$ and the bell center, initially taken as $(\lambda_c, \theta_c) = (3\pi/2 - \Delta\lambda/2, 0)$. The bell radius $R$ is $7\pi/64$ as in Rasch (1994) and NCS99. The exact solution has the same geometry as the initial condition. Accuracy is measured as relative error in the $l_1$, $l_2$, and $l_\infty$ norms, and max and min, consistent with Williamson et al. (1992).

The domain consists of a $128 \times 65$ grid where the first and last latitudinal grid lines are actually the south and the north poles, respectively. Thus, there are $128 \times 64$ grid cells spanning the entire spherical domain. The scalar field is initially distributed at the cell centers (cell average). The time step and the value of the maximum wind speed $u_0$ are chosen such that the angular velocity of the rotational flow is $\omega = 2\pi/256$ radians per time step. That means 256 time steps are required for one complete revolution around the globe. When the wind is in the pole-to-pole direction ($\alpha = \pi/2$), the meridional Courant number is $C_\theta = 0.5$.

### b. Results of solid-body rotation

Numerical experiments have been performed for $\alpha = 0$, $\pi/2 - 0.05$ and $\pi/2$ with exact trajectories. Experiments were conducted for intermediate grids generated with either cubic-Lagrange interpolation or the great-circle approach (see section 3). The great-circle approach is potentially more efficient since it uses the known Cartesian coordinate $(X, Y, Z)$; however, the intermediate grid generated with 1D cubic-Lagrange interpolation was found to be more accurate. For the sake of brevity, only the results from the intermediate grid generated with cubic-Lagrange interpolation are included. Three different options of the CCS scheme are compared to the results with others, these options are the monotonic (CCS-M), the positive-definite (CCS-P), and the CCS without any filtering (CCS-N).

The results from solid-body rotation of a cosine bell after one revolution when flow is in the pole-to-pole direction are displayed in Fig. 2 (the contour values varies from 0.1 to 0.9 with uniform increment 0.1). The exact solution is displayed as dashed contours. In Fig. 2, the top-left panel shows the numerical solution with CCS-N and the solution with CCS-P is shown in the top-right panel. The bottom-left panel of Fig. 2 shows the solution with the CCS-P, but the rotation parameter $\alpha$ is set to $\pi/2 - 0.05$ so that the center of the cosine bell deviates from the poles to avoid symmetry (offset polar flow). The numerical solution with monotonic option, CCS-M is shown in bottom-right panel of the Fig. 2. No smoother or filter is applied to suppress the noise generated at the poles. Table 1 shows the global error as a function of rotation parameter $\alpha$ for the CCS simulations.

For the polar flow, the cosine bell has undergone a small stretching along the flow direction, and the *height* of the cosine bell is slightly reduced as compared to the exact solution. The monotonic constraint has resulted in slight degradation of the shape of the cosine bell, particularly at the center (Fig. 2, bottom-right panel). Also, error is larger for the CCS-M, as compared to the CCS-N or CCS-P (Table 1). For a comparison, the results with cell-integrated semi-Lagrangian (CISL) schemes of NM02 is included in Table 1. The $l_1$ and $l_2$ errors are smaller for the CCS schemes as compared to

CCS−N, Pole−Flow                                    CCS−P, Pole−Flow

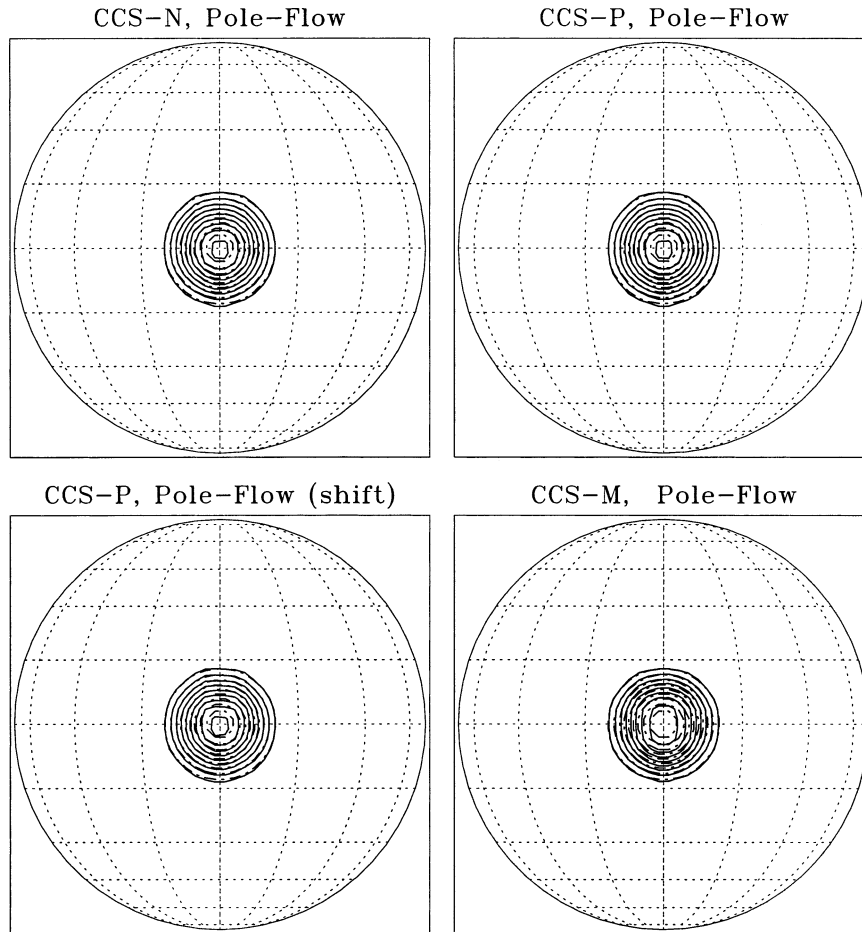CCS−P, Pole−Flow (shift)                            CCS−M,  Pole−Flow

FIG. 2. Results on an orthographic projection for solid-body rotation of a cosine bell after one revolution (256 time steps) when the flow is along the pole-to-pole direction ($\alpha = \pi/2$). The exact solution is shown as the dashed contours. The conservative cascade scheme with positive-definite option (CCS-P), monotonic option (CCS-M) and without any filter option (CCS-N) are used for advection. Top-left shows the numerical solution with the CCS-N and top-right shows that with the CCS-P. The bottom left and right show numerical solutions with the CCS-P ($\alpha = \pi/2 - 0.05$) and the CCS-M, respectively.

the corresponding CISL scheme, but $l_\infty$ errors are smaller for the CISL scheme. Also, the max and min errors for the CISL schemes are slightly smaller than that of the CCS schemes. For a similar set of experiments with conservative schemes, Lin and Rood (1996) and NM02 have reported the slight degradation of numerical solution with monotonic filter. Lin and Rood (1996, see their Table 6, p. 2060) have compared the accuracy of

TABLE 1. Error as the function of rotation angle $\alpha$ for solid-body rotation of a cosine bell after one revolution using the conservative cascade schemes ($\alpha = \pi/2$ corresponds to the flow along pole-to-pole direction). For comparison, results with CISL schemes (NM02) are also included. The leters N, P, and M denote no filter, positive-definite, and monotonic options, respectively. Note that the values of error are set to 0.0 where they reflect round-off errors.

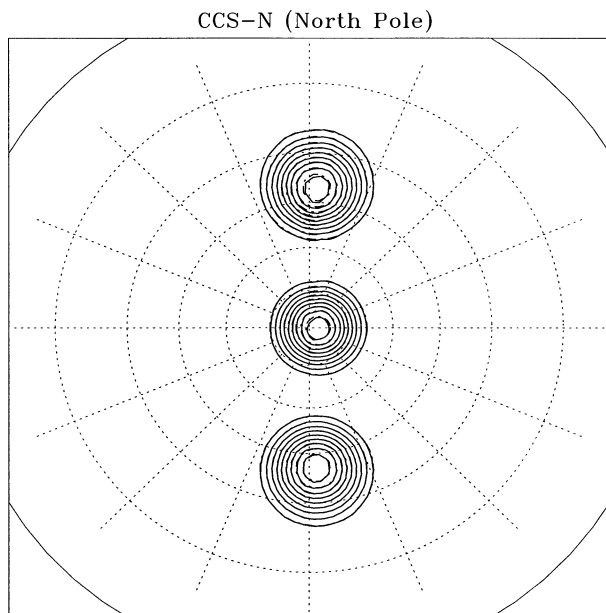| $\alpha$ | Scheme | $l_1$ | $l_2$ | $l_\infty$ | Max | Min |
|---|---|---|---|---|---|---|
| $\pi/2$ | CCS-N | 0.054 | 0.042 | 0.065 | −0.055 | −0.0069 |
| $\pi/2$ | CCS-P | 0.051 | 0.041 | 0.065 | −0.055 | −0.0066 |
| $\pi/2$ | CCS-M | 0.076 | 0.082 | 0.129 | −0.125 | 0.0 |
| $\pi/2 - 0.05$ | CCS-P | 0.055 | 0.043 | 0.064 | −0.049 | −0.0068 |
| 0 | CCS-P | 0.036 | 0.034 | 0.042 | −0.042 | 0.0 |
| $\pi/2$ | CISL-N | 0.063 | 0.046 | 0.048 | −0.016 | −0.0041 |
| $\pi/2$ | CISL-P | 0.059 | 0.045 | 0.048 | −0.016 | −0.0034 |
| $\pi/2$ | CISL-M | 0.084 | 0.084 | 0.109 | −0.052 | −0.0001 |

CCS−N (North Pole)



FIG. 3. Polar stereographic projection for the offset ($\alpha = \pi/2 - 0.05$) solid-body rotation of a cosine bell crossing over the North Pole, at time steps 32, 64, and 96 (starting from bottom to top of the frame). The CCS-N is used for the numerical integration, and the exact solutions are shown as dashed contours.

their conservative and positive-definite scheme flux-form semi-Lagrangian-5 (FFSL-5) with Rasch (1994), for the solid-body rotation ($\alpha = \pi/2$). Results with the CCS-P (Table 1) shows that errors are slightly higher than that of FFSL-5, but much better than that of Rasch (1994) for the same experiment. For example, the $l_1$, $l_2$, and $l_\infty$ errors for FFSL-5 are 0.047, 0.041, and 0.053; for the CSS-P, these values are 0.051, 0.041, and 0.065, respectively.

The most challenging aspect of a solid-body rotation experiment on a sphere is the accuracy of simulation of cross-polar advection. Figure 3 shows a polar stereographic projection of the cosine-bell advection for the offset polar flow ($\alpha = \pi/2 - 0.05$) with the CCS-N as the bell approaches, passes over and leaves the North Pole, at time steps 32, 64, and 96 respectively. There is a small distortion for the numerical solution when the pattern crosses the pole; however, after one revolution the numerical solution appears very similar to the case when $\alpha = \pi/2$, as shown in Fig. 2, top-left panel.

The results with the CCS scheme are comparable to that of the CISL scheme (NM02), however, the CCS scheme has additional advantages. The CISL scheme (Machenhauer and Olk 1998; NM02) requires global information of the coefficients of the quasi-biparabolic functions involved in the remapping algorithm. This potentially increases the memory requirements as the coefficients must be stored. Since the CCS remapping needs only a few operations (as compared to the CISL remapping), it is more computationally efficient than the CISL scheme.

The computational efficiency of the CCS has been compared with the CISL scheme, and a standard (non-conservative) SL scheme combined with bicubic-Lagrange interpolation (SL-BCL) is used for a reference. The solid-body rotation of a symmetric slotted-cylinder in 2D Cartesian geometry, is examined with after one revolution with parameters identical to those used in the experiments of Nair et al. (1999b). The CCS with linearly interpolated intermediate grid and monotonic filter are found to be 2.4 times more efficient than SL-BCL and about 2.1 times more efficient than CISL with monotonic filter. For a similar experiment, the conservative piecewise biparabolic method introduced by Rančić (1992) is 2.5 times more expensive than the SL-BCL scheme. In spherical geometry, computational efficiency of CCS without any filter is compared with that of SL-BCL. An intermediate grid for CCS scheme is generated with cubic-Lagrange interpolation and additional points are used around the polar zones as described earlier. In this case, CCS is found to be approximately 2 times as efficient as SL-BCL for solid-body rotation around the globe ($\alpha = \pi/2$ with 256 time steps).

### c. Deformational flow test

To examine the accuracy and robustness of the CCS scheme, we consider two deformational flow (cyclogenesis) tests in spherical geometry. NCS99 gives the details of the idealized cyclogenesis problem of Doswell (1984) on the surface of the sphere. In the first test, a flow with smoothly varying positive-definite field is used (NM02) for advection. The second test is the nonsmooth flow problem as given in NCS99. For these experiments, the intermediate grid is generated by using the 1D cubic-Lagrange interpolation.

#### 1) SMOOTH DEFORMATIONAL FLOW

Here the vortex problem considered by NM02 is used. There are two vortices in the domain, one near each pole, and the CCS positive-definite is used for transport. A nonconservative SL scheme with bicubic-Lagrange interpolation is used for comparison (NCS99).

Let ($\lambda'$, $\theta'$) be the rotated spherical coordinate system with respect to the regular spherical coordinate system ($\lambda$, $\theta$), such that the north pole of the rotated sphere is located at ($\lambda_0$, $\theta_0$) in ($\lambda$, $\theta$) system. Then two circular vortices may be created with centers at the poles of the ($\lambda'$, $\theta'$) system. Normalized tangential velocity of the vortex is defined as

$$V_t = \frac{3\sqrt{3}}{2} \operatorname{sech}^2(\rho') \tanh(\rho'), \tag{24}$$

where $\rho' = r_0 \cos\theta'$ is the radial distance of the vortex and $r_0$ is a constant. The angular velocity $\omega'(\theta')$ is defined as (NM02),
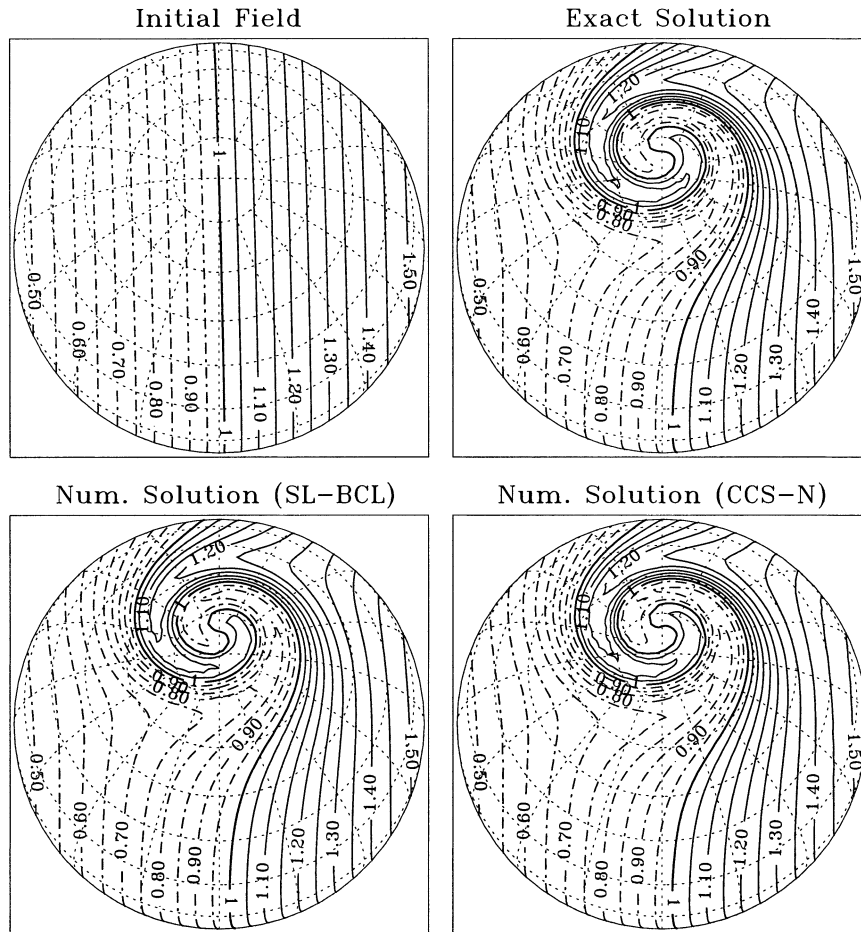
FIG. 4. Results on an orthographic projection for the polar vortex simulations (smooth deformational flow). Top-left shows the initial flow field and top-right shows exact solution at $t = 3$. The vortex center is located near the pole. Numerical solutions are produced by using the CCS-N (bottom-right) and the SL scheme with bicubic Lagrange interpolation (SL-BCL, bottom-left).

$$\omega'(\theta') = \begin{cases} 0 & \text{if } \rho' = 0, \\ \dfrac{V_t}{\rho'} & \text{if } \rho' \neq 0. \end{cases} \quad (25)$$

The exact solution at time $t$ is

$$\psi(\lambda', \theta', t) = 1 - \tanh\left[\frac{\rho'}{d} \sin(\lambda' - \omega't)\right], \quad (26)$$

where $d$ is the parameter that defines the characteristic width (NCS99).

With these conditions two symmetric vortices (with respect to the equator of the rotated sphere) can be formed. The vortex centers are approximately located at 81°N and 81°S latitudes, respectively, if the parameters values are chosen to be $(\lambda_0, \theta_0) = (\pi + 0.025, \pi/2.2)$, $r_0 = 3$ and $d = 5$. This configuration keeps the vortex centers away from the poles of the $(\lambda, \theta)$ system, to avoid possible symmetry.

The initial values of the field (or cell averages) are generated at the cell centers on a $128 \times 64$ grid, and integrated for 3 time units (nondimensional) with 64 time steps. The corresponding Courant number in $\lambda$, $\theta$ directions are $C_\lambda = 19.1$ and $C_\theta = 0.82$, respectively. The exact upstream positions and other details of the experiment are given in NM02. Measures of error for the polar vortex problem are the same as that for the solid-body rotation test.

Orthographic projection of the smooth deformational flow simulation are shown in Fig. 4, where contour values range from 0.5 to 1.5. The flow fields are visually indistinguishable for the CCS-N and CCS-P, and the corresponding error measures are almost identical. Figure 4 shows the initial and the exact solution at the top left and right panels, respectively, while the bottom panel shows the numerical solution with SL-BCL without any filter, and the bottom-right panel shows the numerical solution with the CCS-N. The center of the vortex is captured well in the CCS-N simulation and is very similar to the exact solution. Moreover, the vortex center is more accurately represented in the CCS-N simulation as compared to that of the SL-BCL. Table 2

TABLE 2. Error at $t = 3.0$ for the smooth deformational flow problem (polar vortices) using the conservative cascade schemes and a nonconservative semi-Lagrangian scheme combined with bicubic Lagrange interpolation (SL-BCL).

| Scheme | $l_1$ | $l_2$ | $l_\infty$ | Mass | Max | Min |
|--------|-------|-------|------------|------|-----|-----|
| CCS-P | 0.0011 | 0.0027 | 0.0154 | 0.0 | 0.000136 | −0.000188 |
| CCS-M | 0.0013 | 0.0033 | 0.0220 | 0.0 | −0.000016 | −0.000073 |
| SL-BCL | 0.0072 | 0.0143 | 0.0585 | 0.000241 | 0.0 | 0.0 |

shows the error for different versions of the CCS and the SL-BCL. The $l_1$, $l_2$, and $l_\infty$ errors are smaller for the CCS as compared to the SL-BCL. However, max and min errors for the CCS are slightly different from the zero errors of the SL-BCL.

### 2) NONSMOOTH DEFORMATIONAL FLOW

The robustness of the CCS is further demonstrated by using the nonsmooth vortex (cyclogenesis) problem considered by NCS99. Here the vortex is defined on a polar-stereographic plane tangent to the north pole of the rotated coordinate system $(\lambda', \theta')$ (NCS99). The results of the vortex simulation is projected onto the tangent plane, rather than using orthographic projection as in the previous case.

The exact solution at time $t$ is given as

$$\psi(\lambda', \theta', t) = -\tanh\left[\frac{\rho'}{d}\sin(\lambda' - \omega't)\right], \quad (27)$$

where $\rho'$ is the distance from the pole of projection to a point $(\lambda', \theta')$ in the tangent plane, and $d$ defines the characteristic width of the frontal zone. Specific details of other parameters of projection, exact trajectory, and derivation velocity filed are given in NCS99 and will not be discussed here.

The vortex center is located approximately 65°N and the isopleth of maximum wind speed passes over the north pole of the computational domain. This makes the problem more challenging, since the polar Courant numbers are higher as compared to solid-body rotation tests considered. The smoothness parameter for this problem is set to $d = 0.01$ (a much smaller value than that used in the smooth deformational flow) and $t = 2.5$ time units are used to create the exact solution. The computational domain consists of $128 \times 64$ cells as in the previous cases, and the values of the initial scalar field $\psi(\lambda', \theta', 0)$ at the cell centers are used as the initial cell averages. The monotonic version of the conservative cascade scheme is used with 64 time steps equivalent to $t = 2.5$ time units. Figures 5a and 5b show the exact solution projected on to a tangent plane, at initial time and at $t = 2.5$ time units respectively. The numerical solution at $t = 2.5$ produced by using the CCS-M is shown in Fig. 6.

The numerical solution shown in Fig. 6, with CCS-M is very similar to that with the monotonic cascade-spline SL shown in Fig. 8b of NCS99. The $l_1$, $l_2$, and $l_\infty$ errors for CCS-M are 0.035, 0.13, and 1.06, respec-

tively, and the corresponding errors for monotonic cascade-spline SL in NCS99 are slightly better. The max and min error measures for CCS-M are 0.0052 and −0.0029; for the monotonic cascade-spline SL scheme (NCS99) these values are zero. Note that, in NCS99, a nonconservative SL scheme with cascade cubic-spline interpolation is used for vortex simulation with $128 \times 65$ grid points in the computational domain. This experimental setup is slightly different from the present one, and therefore not strictly comparable.

A reason for generating small spurious values for the monotonic cases could be the rounding error associated with the subtraction [see Eq. (13)]. In Eq. (13), the mass at the new time level is estimated by subtracting the accumulated masses obtained *after* applying the monotonic filter. Also, max and min are generated in the immediate vicinity of the quasi discontinuity of the solution. Since the monotonic filter is not completely removing the spurious values at every time step and the wind field used is uniform, the error accumulates during the integration. The creation of tiny spurious values even after applying monotonic filter is reported in Lin and Rood (1996) and NM02 for their respective conservative schemes. A 1D monotonic short-wave filter applied after remapping operation could remove the noise.

### 5. Summary and conclusions

An efficient mass-conservative 2D advection scheme appropriate for global applications is presented. The scheme combines a mass-conservative finite-volume advection with semi-Lagrangian method in a computationally efficient cascade interpolation framework. This has been tested previously (R95), but only for Cartesian coordinates. Following Machenhauer and Olk (1996, 1998), the extension to spherical geometry is facilitated by performing the integrations over Lagrangian cells, which are approximated by polygons with sides parallel to the coordinate axis in a Cartesian $(\lambda, \mu = \sin\theta)$ coordinate system (where $\lambda$ is longitude and $\theta$ is latitude). As in R95 the accurate piecewise parabolic representation of Colella and Woodward (1984) is used. Over the polar caps, the areas near the poles where the conservative cascade procedure breaks down, we have introduced the globally conservative, but locally approximate, scheme as one used in NM02. This procedure is currently restricted to polar meridional Courant numbers less than one. The resulting conservative cascade scheme (CCS) is demonstrated to be accurate for the
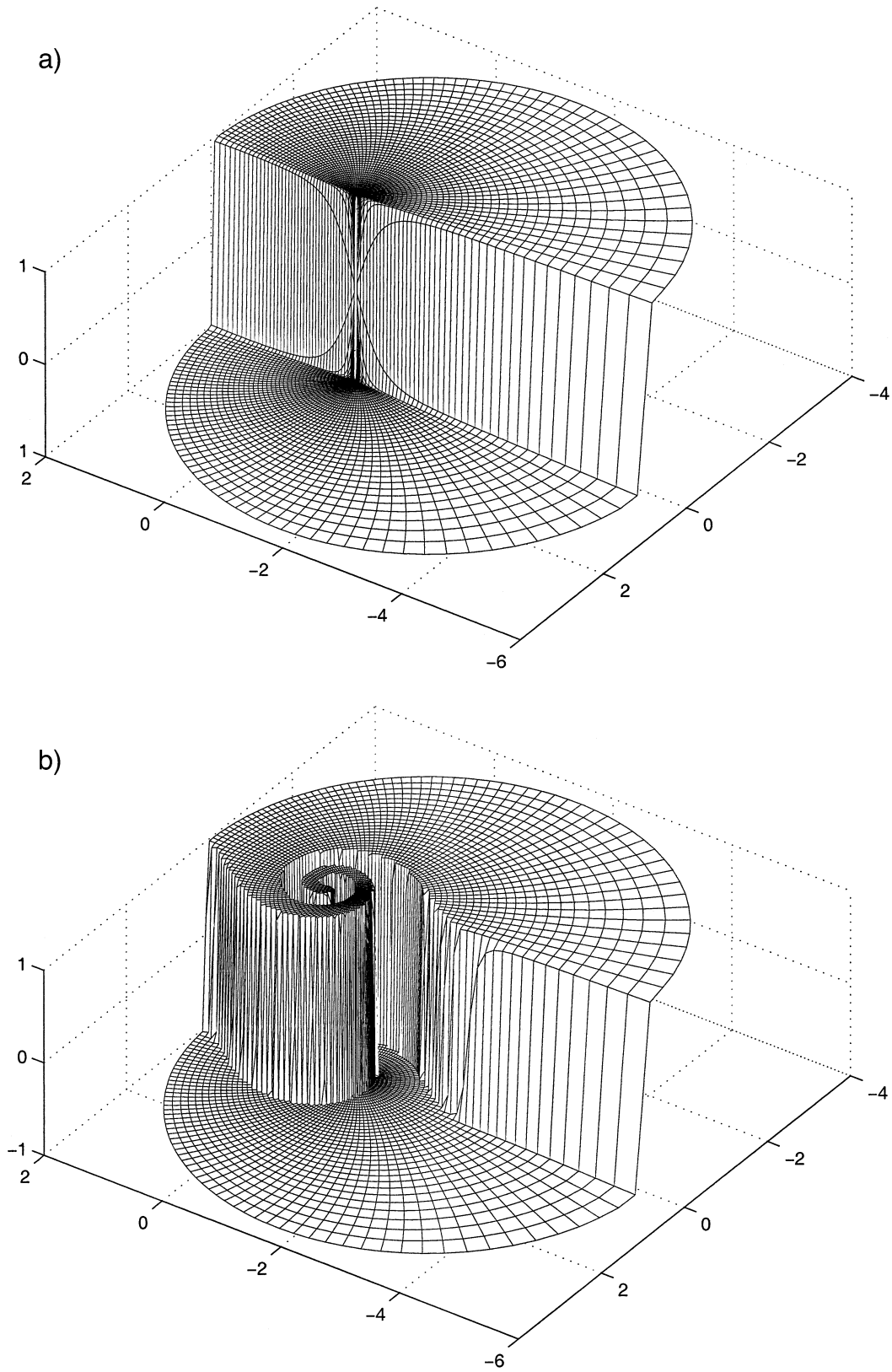
a)



b)



Fig. 5. (a) Exact solution of the nonsmooth deformational flow problem (see text), projected onto a plane tangent to the vortex center at initial time. (b) Exact solution of the nonsmooth deformational flow problem (see text), projected onto a plane tangent to the vortex center at time $t = 2.5$ units.
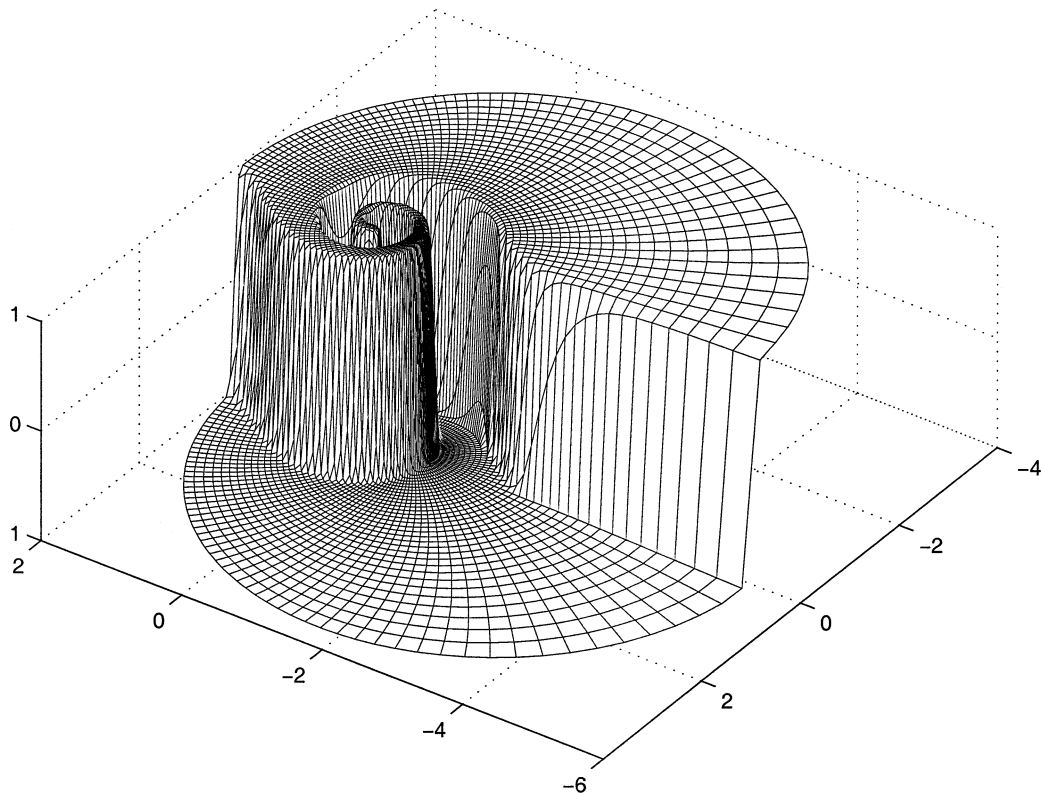
Fig. 6. Numerical solution of the nonsmooth deformational flow problem at $t = 2.5$ units (64 time steps), projected onto a plane tangent to the vortex center. The conservative cascade scheme with monotonic filter (CCS-M) and with exact trajectories are used for the numerical integration.

tests of solid-body rotation of a coherent structure and deformational flow over the poles.

The simple redistribution of mass enclosed in the polar cap to the constituent cells, using the SL method (NM02), is found to be accurate. The numerical simulation of cross-polar transport is accurate and numerical solutions are comparable to the accurate results obtained in similar conservative advection experiments in Lin and Rood (1996). The intermediate grid generation methods, using cubic interpolation of the Lagrangian coordinates and the great-circle approach, are found to be effective for conservative transport. However, the cubic interpolation method is more accurate while the great-circle approach is more computationally efficient.

Computational efficiency of the conservative cascade scheme is compared with a standard bicubic-Lagrange interpolation SL scheme and the CISL scheme (Machenhauer and Olk 1997; NM02). In the present implementations CCS is found to be twice as efficient as our implementation of a full 2D SL scheme and more than twice as efficient as the CISL scheme. Results produced with the CCS are comparable to that of the CISL scheme; however, the memory requirement for the CCS is less than that of the CISL scheme, at least in the present implementations. Computational efficiency of the CCS is evident even without optimizing the code.

In 3D, with multiple tracer fields used as in typical global chemistry models, the CCS can save a significant amount of computer time while being remaining very accurate.

The monotonic and positive-definite filter options are found to be very effective for removing the overshoots and undershoots of the numerical solution. However, the filter constraints are applied before the remapping [see Eq. (13)], and this may result in creating small new spurious values due to truncation error. A 1D conservative monotonic (or positive-definite) filter may be used after each cascade cycle to obviate this problem.

The CCS approach presented is valid for meridional Courant number $C_\theta \leq 1$, and it is independent of zonal Courant number $C_\lambda$. Therefore, the CCS can be used as an efficient semi-Lagrangian option for any global chemistry–climate Eulerian models. However, for a more general SL application ($C_\theta > 1$), the special polar treatment developed by NM02 may be used for the CCS. Generalizing CCS without Courant number restriction and application to realistic 3D models are beyond the scope of present study. Work is in progress to implement the CCS into a global nonhydrostatic SL model (Qian et al. 1998) developed at North Carolina State University.

## REFERENCES

Carpenter, R. L., K. K. Droegemeier, P. R. Woodward, and C. E. Hane, 1990: Application of piecewise parabolic method (PPM) to meteorological modeling. *Mon. Wea. Rev.,* **118,** 586–612.

Chorin, A. J., and J. E. Marsden, 1993: *A Mathematical Introduction to Fluid Mechanics.* 3d ed. Springer-Verlag, 169 pp.

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

Côté, J., S. Gravel, A. Methot, A. Patoine, M. Roch, and A. Staniforth, 1998: The operational CMC/MRB global environmental multiscale (GEM) model. *Mon. Wea. Rev.,* **126,** 1373–1395.

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

Dukowicz, J. K., 1984: Conservative rezoning (remapping) for general quadrilateral meshes. *J. Comput. Phys.,* **54,** 411–424.

——, and J. R. Baumgardner, 2000: Incremental remapping as a transport/advection algorithm. *J. Comput. Phys.,* **160,** 318–335.

Hourdin, F., and A. Armengaud, 1999: The use of finite-volume methods for atmospheric advection of trace species. Part I: Test of various formulation in a general circulation model. *Mon. Wea. Rev.,* **127,** 822–837.

Laprise, J. P. R., and A. Plante, 1995: A class of semi-Lagrangian integrated mass (SLIM) numerical transport algorithms. *Mon. Wea. Rev.,* **123,** 553–565.

Leslie, L. M., and R. J. Purser, 1995: Three dimensional mass-conserving semi-Lagrangian schemes employing forward trajectories. *Mon. Wea. Rev.,* **123,** 2551–2566.

Li, Y., and J. S. Chang, 1996: A mass-conservative, positive-definite, and efficient Eulerian advection scheme in spherical geometry and on a nonuniform grid system. *J. Appl. Meteor.,* **35,** 1897–1913.

Lin, S.-J., and R. B. Rood, 1996: Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Wea. Rev.,* **124,** 2046–2070.

Machenhauer, B., and M. Olk, 1996: On the development of a semi-Lagrangian cell integrated shallow water model on the sphere. *Proc. ECMWF Workshop on Semi-Lagrangian Methods,* Reading, United Kingdom, ECMWF, 231–228. [Available from ECMWF, Shinfield Park, Reading, Berkshire RG2 9AX, United Kingdom.]

——, and ——, 1997: The implementation of the semi-implicit scheme in cell-integrated semi-Lagrangian models. *Atmos.–Ocean,* **35,** 103–126.

——, and ——, 1998: Design of a semi-implicit cell integrated semi-Lagrangian model. Max Planck Institute for Meteorology Tech. Rep. 265, 76–87. [Available from Max Planck Institute for Meteorology, Bundesstrasse-55, D-20146 Hamburg, Germany.]

Moorthi, S., R. W. Higgins, and J. R. Bates, 1995: A global multilevel atmospheric model using a vector semi-Lagrangian finite-difference scheme. Part II: Version with physics. *Mon. Wea. Rev.,* **123,** 1523–1541.

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

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

——, ——, and ——, 1999b: Monotonic cascade interpolation for semi-Lagrangian advection. *Quart. J. Roy. Meteor. Soc.,* **125,** 197–212.

Prather, M. J., 1986: Numerical advection by conservation of second order moments. *J. Geophys. Res.,* **91,** 6671–6681.

Priestley, A., 1993: A quasi-conservative version of the semi-Lagrangian advection scheme. *Mon. Wea. Rev.,* **121,** 621–629.

Purser, R. J., and L. M. Leslie, 1991: An efficient interpolation procedure for high-order three-dimensional semi-Lagrangian models. *Mon. Wea. Rev.,* **119,** 2492–2498.

Qian, J.-H., F. H. M. Semazzi, and J. S. Scroggs, 1998: A global nonhydrostatic semi-Lagrangian atmospheric model with orography. *Mon. Wea. Rev.,* **126,** 747–771.

Rančić, M., 1992: Semi-Lagrangian piecewise biparabolic scheme for two-dimensional horizontal advection of a passive scalar. *Mon. Wea. Rev.,* **120,** 1394–1406.

——, 1995: An efficient, conservative, monotonic remapping for semi-Lagrangian transport algorithms. *Mon. Wea. Rev.,* **123,** 1213–1217.

Rasch, P. J., 1994: Conservative shape-preserving two-dimensional transport on a spherical reduced grid. *Mon. Wea. Rev.,* **122,** 1337–1350.

——, 1998: Recent development in transport schemes at NCAR. Max Planck Institute for Meteorology Tech. Rep. 265, 65–75. [Available from Max Planck Institute for Meteorology, Bundesstrasse-55, D-20146 Hamburg, Germany.]

Scroggs, J. S., and F. H. M. Semazzi, 1995: A conservative semi-Lagrangian method for multidimensional applications. *Numer. Methods Partial Differential Equations,* **11,** 445–452.

Smolarkiewicz, P. K., and P. J. Rasch, 1991: Monotone advection on the sphere. *J. Atmos. Sci.,* **48,** 793–810.

——, and J. A. Pudykiewicz, 1992: A class of semi-Lagrangian approximations for fluids. *J. Atmos. Sci.,* **49,** 2082–2096.

Staniforth, A., and J. Côté, 1991: Semi-Lagrangian integration schemes for the atmospheric models—A review. *Mon. Wea. Rev.,* **119,** 2206–2223.

Williamson, D. L., and J. G. Olson, 1998: A comparison of semi-Lagrangian and Eulerian polar climate simulations. *Mon. Wea. Rev.,* **126,** 991–1000.

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